Interpretable and Causal Analysis for Multivariate Time Series

Amin Dhaou

INSTITUT POLYTECHNIQUE DE PARIS

NNT : 2024IPPAX037

Thèse de doctorat

ÉCOLE POLYTECHNIQUE

IP PARIS

Interpretable and Causal Analysis for Multivariate Time Series

Thèse de doctorat de l'Institut Polytechnique de Paris
préparée à l'École polytechnique

École doctorale n°574 École doctorale de mathématiques Hadamard (EDMH)
Spécialité de doctorat : Mathématiques appliquées

Thèse présentée et soutenue à Palaiseau, le 04 juillet 2024, par

AMIN DHAOU

Composition du Jury :

Rémi Flamary Professeur, École polytechnique (CMAP)	Président
Souhaib Ben Taieb Associate Professor, University of Mons (UMONS)	Rapporteur
Marianne Clausel Professeure, Universite de Lorraine (IECL)	Rapporteur
Emilie Devijver CNRS researcher, Université Grenoble Alpes (LIG)	Examinatrice
Erwan Scornet Professeur, Sorbonne Université (LPSM)	Examinateur
Josselin Garnier Professeur, École polytechnique (CMAP)	Directeur de thèse
Erwan Le-Pennec Professeur, École polytechnique (CMAP)	Directeur de thèse
Louis Verny Ingénieur de recherche, TotalEnergies OneTech (R&D Power)	Invité

Acknowledgements

First and foremost, I would like to express my deepest gratitude to God for giving me the strength, guidance, and patience throughout this journey. I am thankful for the many blessings that helped me overcome challenges and stay committed to this work. Completing this thesis, with its ups and downs, has been a valuable learning experience. For everything, I am sincerly grateful.

I want to take this opportunity to thank my parents for their unwavering support and encouragement. They have instilled in me the values of hard work, perseverance and dedication, which have been the foundation of everything I have achieved. I owe so much of my success to them, and this accomplishment belongs to them as much as it does to me.
To my brother and sisters, thank you for always being a source of positivity and motivation. Your presence and energy have been a great support throughout this thesis, and I am so grateful to have you by my side.
A special thanks to my wife, whose constant support, encouragement, and remarkable patience over these last months have been priceless. Your presence has made this achievement possible.
Lastly, my sincere thanks to my family and friends for their continued support along this path.

I would like to thank the École Polytechnique and TotalEnergies, with whom this PhD thesis was conducted as part of a CIFRE collaboration, in partnership with the SINCLAIR laboratory.

I would also like to extend my deepest thanks to my thesis supervisors. Josselin, thank you for giving me the opportunity to embark on this journey. Your dedication, availability, support, humility and kindness have deeply inspired me, and I am truly grateful.

Erwan, I also want to thank you for your invaluable mentorship. Your kindness, patience, clarity, and ability to explain complex ideas made this experience much enriching. The support from you and Josselin gave me the confidence to keep pushing forward.

A special thanks to Antoine, who initiated this thesis and made it possible for me to join TotalEnergies. Your support at the beginning laid the foundation for everything that followed.
I also want to thank Louis. Even though you came in towards the end, your willingness to help and the valuable discussions we had were highly beneficial.
I am deeply thankful to both institutions, as well as to all the colleagues and professors, for their intellectual contributions, without which this work would not have been possible.

I am grateful to Souhaib and Marianne for reviewing this work carefully and providing

constructive feedback that helped improve the thesis. I also want to thank the jury members, Emilie, Rémy, and Erwan, for their insightful questions and observations.

My sincere thanks go to all the doctoral students at TotalEnergies and CMAP. The time spent with you made this journey not only enjoyable but also filled with great memories. A special mention to Naoufal, Cheikhna, Benjamin, Baptiste, Ali, Yagnik, Wassil, Elie, Khalid, Yanis, Mehdi, Amar, Mohammed, Hossein, and to all those who were part of this experience for all the meaningful conversations and the moments we shared.

At École Polytechnique, I would like to thank Vincent, Ali, Guillaume, Corentin, Charles, Constantin, Mehdi, Raphael, Paul, Anouar, Antoine, Armand, Benjamin, Charu, and all the other PhD students at CMAP. The memories we created, especially during conferences and trips will always stay with me.

Finally, I am deeply grateful to all my teachers and everyone I have met throughout my life who, in one way or another, contributed to where I am today.

For those preparing to defend, I wish you the very best in your upcoming defenses, and to everyone else, I wish you continued success in all your future endeavors.

Résumé

Les avancées en intelligence artificielle ont permis le développement de modèles de plus en plus complexes permettant de résoudre de nombreuses tâches. Dans des domaine d'applications critiques tels que l'industrie ou la médecine, il s'avère nécessaire de proposer des modèles dit interpretables établissant clairement le mécanisme de décisions favorisant ainsi la compréhension de ces modèles et de leurs décisions par les utilisateurs, et par conséquent leur acceptation. Ces objectifs relèvent du domaine de l'Intelligence Artificielle eXplicable (XAI), qui connaît un intérêt croissant depuis quelques années.

Les données de séries temporelles, qui mesurent l'évolution de variables au fil du temps, comme les relevés de capteurs, fournissent des informations précieuses sur le comportement des systèmes. En identifiant des structures dans ces données, nous pouvons comprendre les interactions entre les variables, améliorer la précision des prévisions et concevoir de meilleures stratégies d'intervention. Cette thèse étudie l'analyse de données de séries temporelles à haute dimension en se concentrant sur l'explication des déviations de systèmes par rapport à leur fonctionnement normal et sur la modélisation de la dynamique sous-jacente de systèmes permettant de prédire leur évolution.

Ce travail a deux objectifs principaux. **Le premier objectif** est de développer un algorithme interprétable qui identifie les causes racines des comportements normaux et anormaux dans les données de séries temporelles. Diverses techniques sont utilisées pour identifier les causes racines, mais elles présentent des limites quant à leur capacité à traiter de grandes dimensions et à distinguer la causalité des corrélations. Une approche basée sur le concept de causalité de Granger [Granger 1988], qui extrait des relations interprétables et causales sous la forme de règles, a été développée pour remédier à ces limitations. L'algorithme qui en résulte est conçu pour traiter différents types de données (numériques, catégorielles), pour fournir aux utilisateurs des explications interprétables du problème et pour développer des règles prédictives permettant de désamorcer les phénomènes anormaux à l'avance.

Le deuxième objectif vise à développer un modèle de prévision qui non seulement prédit les valeurs futures, mais extrait également la dynamique sous-jacente des séries temporelles influençant ces prédictions. Ce domaine appelé régression symbolique favorise la transparence pour les utilisateurs en expliquant le raisonnement du modèle. Les modèles de régression avec pénalisation parcimonieuse sont largement utilisés dans ce domaine pour leur capacité à apprendre des dynamiques complexes dans des scénarios de grande dimension. Néanmoins, leurs performances en matière de prévision peuvent être limitées, en particulier pour des données complexes et non linéaires. Pour y remédier, nous proposons une nouvelle approche

qui combine la régression pénalisée et la correction des erreurs dans un cadre de prévision des séries temporelles afin d'améliorer l'apprentissage des dynamiques sous-jacentes. En outre, le modèle est conçu pour traiter des données de séries temporelles complexes et non linéaires.

En atteignant ces objectifs, cette recherche a le potentiel d'améliorer de manière significative notre capacité à analyser et à comprendre les données de séries temporelles. Il en résultera de meilleures prévisions, une meilleure compréhension du système et le développement de stratégies d'intervention plus efficaces.

Abstract

Advances in artificial intelligence have led to the development of increasingly complex models for solving a wide range of tasks. In critical applications such as industry and medicine, it has become necessary to propose "interpretable" models that clearly establish the decision-making process, thus promoting understanding of these models and their decisions and, consequently, their user acceptance. These objectives fall within the field of eXplainable Artificial Intelligence (XAI), which has been attracting growing interest in recent years.

Time-series data, which measure the evolution of variables over time, such as sensor readings or data monitoring, provide valuable information on the system's behavior. By identifying patterns in these data, we can understand the interactions between variables, improve forecasting accuracy, and design better intervention strategies. This thesis studies the analysis of high-dimensional time-series data, focusing on explaining local system deviations from normal operation and, on the global scale, modeling the underlying dynamics of the system to predict its evolution.

This work has two main objectives. **The first objective** is to develop an interpretable algorithm that identifies the root causes of both normal and abnormal behavior in time series data. Various techniques are used to identify root causes, but they suffer from limitations in their ability to handle high dimensions and to distinguish causality from correlations. To overcome these limitations, an approach based on the concept of Granger causality [Granger 1988], which extracts interpretable and causal relationships in the form of rules, has been developed. The resulting algorithm is designed to handle different data types (numerical, categorical), provide users with interpretable explanations of the problem, and develop predictive rules to defuse the event in advance.

The second objective aims at developping a forecasting model that not only predicts future values but also reveals the underlying dynamic of the time series influencing those predictions. This field, called symbolic regression, fosters transparency for users by explaining the model's reasoning. Regression models with parsimonious penalization are widely used in this field for their ability to learn complex dynamics in high-dimensional settings. Nevertheless, their forecasting performances can be limited, especially for complex, non-linear data. To address this, we propose a novel approach that combines penalized regression with forecasting error correction within a time series forecasting framework for improved learning of underlying dynamics.

By achieving these goals, this research has the potential to significantly improve our ability to analyze and understand time series data. This will result in better forecasts, a better

understanding of the system, and the development of more effective intervention strategies.

Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

Contents

1.1 Context

The early 1960s witnessed the emergence of Artificial Intelligence (AI) as a distinct field. Pioneering researchers, Allen Newell and Herbert Simon, aimed to replicate human problem-solving capabilities through the development of the "Logic Theorist" program based on logic (Russell & Norvig, 2010). The early models, based on logic operators, symbolic reasoning, tree structure, and rule-based systems, led to the development of computer programs like logic rules, expert systems, and decision trees. These models were relatively simple, allowing for easy comprehension of decision-making processes. While these models have proved promising in specific domains, their limitations became apparent when confronted with more complex challenges. Concurrently, efforts were made to develop more complex approaches, including early neural networks like the perceptron. However, these attempts were limited by various technical constraints, ultimately hindering their success.

Since the late 20th century, significant computer hardware advancements have driven significant increases in their computational power. This, coupled with the simultaneous development of data storage capacity, has enabled the collection and storage of large amounts of data. These technical developments, combined with rapid progress in algorithms and mathematical research, have laid the groundwork for the popularization of artificial intelligence.
Since the turning point of AI was in ImageNet Competition in 2012, with the success of deep learning in computer visions tasks (Krizhevsky et al., 2012), machine learning has undergone significant progress, yielding increasingly efficient models in diverse tasks, from decision-making and prediction to forecasting. Its reach extends beyond powering conversational AI through natural language processing, computer vision for image and video recognition and generation, speech analysis, and recommender systems. As a result, machine learning applications have permeated every sector, including healthcare, finance, industry, automotive, and marketing. Indeed, adopting AI has become strategically imperative for businesses to remain competitive and efficient.

While advancements have allowed machine learning models to have impressive accuracy and efficiency, they have also yielded increasingly complex systems. Often referred to as "**black boxes**" or "opaque" models -models with complex/unknown internal workings where only inputs/outputs are observed - as opposed to **transparent** models, their designs hinder understanding and justifications of their decisions when applied in a real-world environment.

The opacity of AI models amplifies ethical considerations around bias in algorithms during learning and accountability for decisions, particularly in critical domains like healthcare (Morley et al., 2020), autonomous vehicles (Martinho et al., 2021), recruitment (Hofeditz et al., 2022), and chatbots (Wiltz, 2017). This lack of transparency hinders our understanding of how the model arrives at its decisions, making it difficult to identify and address potential biases or weaknesses that might lead to errors and discriminatory outcomes. Consequently, opaque models raise ethical and legal concerns for businesses and society.

The widespread adoption of AI, particularly in sensitive areas impacting humans, society, and finances, highly depends on trust and acceptance. This translates to a clear need from different stakeholders (Preece et al., 2018) with three main motivations and demands: transparency, human interaction and trustworthy models (Vilone & Longo, 2021).

The need for understanding AI systems gave rise to Explainable Artificial Intelligence (XAI). This field focuses on developing and analyzing both new and existing machine learning models,

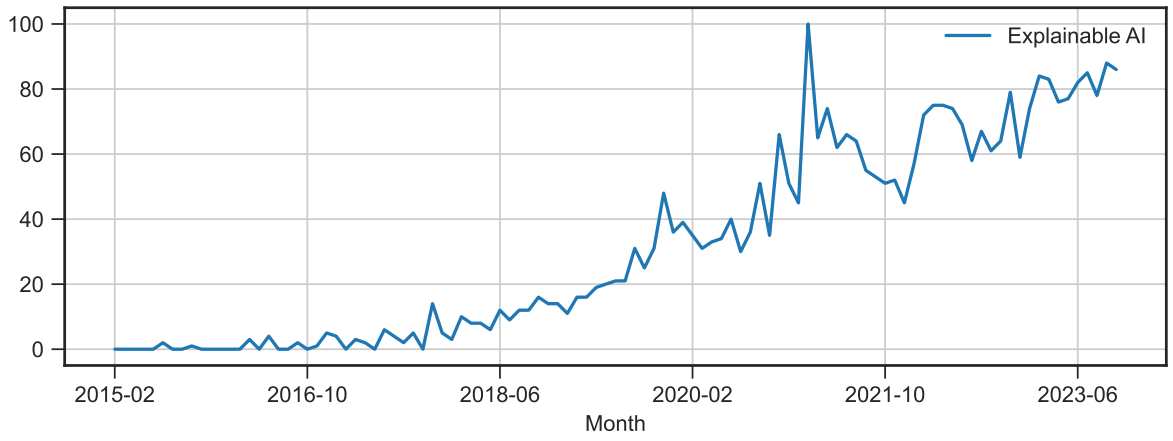


Figure 1.1: Google Trends Index (Max value is 100) of the term “Explainable AI” over the last years (2015–2023).

with the goal of (Barredo Arrieta et al., 2020):

- ❖ uncovering and addressing biases
- ❖ enhancing model robustness
- ❖ providing causal reasoning behind their decisions

1.2 Why do we need explanation

Machine Learning offers a powerful means to enhance the predictive capabilities of systems, enabling them to make informed decisions based on data-driven insights. This is achieved by utilizing various algorithms, each providing a degree of explanation, such as uncertainty, that can serve as an indicator of algorithmic confidence. However, in contexts where the consequences of errors can be severe, such as in critical domains like healthcare or autonomous vehicles, the inherent lack of transparency and the challenge of understanding the decision-making process become a major concern.
The trustworthiness of a machine learning model is intimately tied to the ability to explain its decisions. In these high-stakes scenarios, it is imperative to have models that are not only predictive and accurate but also interpretable. It will allow stakeholders to understand how and why a particular decision has been taken, providing transparency and accountability. In these settings, interpretability becomes an ethical and practical necessity.

In such applications, one will need guarantees not only on the model’s performance but also on its reliability. It is essential to develop methods for rigorously evaluating the performance and robustness under various data conditions and scenarios and the model’s inner workings and decision-making processes to build this trust in machine learning models. Quantitative metrics are typically used for evaluation, but qualitative assessments might be included depending on the problem and objectives.

1.3 Objective of my research

An important aspect of this research involves understanding the changes over time and what causes a system to deviate from normal operating conditions or uncovering the hidden dynamics in time series data. In fact, time series data essentially tracks how variables change over time, such as stock prices, weather data, brain monitoring, or heart rate monitoring. By uncovering patterns in data, we gain insights into how variables interact, leading to improved forecasting, a deeper understanding of system stability, and the development of effective intervention strategies.

Objective 1: Develop an algorithm that finds the root causes of normal/abnormal behaviors in a time series

- ❖ This objective aims to develop an algorithm that provides insights and explanations for deviations from normal behavior.
- ❖ The algorithm should be able to handle various types of time series data, including numerical, categorical, and multivariate data.
- ❖ The explanations should reveal the causes and be understood by humans.

Another key aspect of this research lies in addressing crucial knowledge gaps within time series data analysis. Firstly, we lack a comprehensive understanding of abnormal behaviors' underlying causes and timing. This limitation hinders our ability to anticipate and mitigate potential risks. Secondly, we struggle to explain systems' dynamic behavior and identify the variables driving their evolution. This lack of explanatory power restricts our ability to understand systems and optimize performance.

Objective 2: Develop a forecasting model that uncovers the underlying dynamic of a time series and that predicts future evolution

- ❖ This objective seeks to develop a forecasting model that can not only predict future values in a time series but also explain its predictions.
- ❖ The model should be transparent, allowing users to understand the factors influencing the forecast and gain insights into the underlying trends and patterns.
- ❖ The model should be able to handle complex time series with non-linearities.

1.4 Thesis outline

Chapter 2: Understanding ML models This chapter lays the foundation for our research by establishing a shared understanding of Explainable AI (XAI). We explore this field's challenges, definitions, methods, and evaluations, providing context for the methodologies employed throughout the thesis.

Chapter 3: Analyzing Time Series: Uncovering Patterns and Influencing Factors This chapter delves into root cause analysis, highlighting the main methods and their limitations. It allows us to introduce our proposed solution in Chapter 4.

Chapter 4: Causal and Interpretable Rules for Time Series Analysis This chapter presents our first contribution to root cause analysis. We introduce a novel approach that incorporates both causality and interpretability, addressing the issues identified in Chapter 3.

Chapter 5: Multi-step Ahead Forecasting and Dynamic System Discovery This chapter introduces the domain of forecasting and dynamic system discovery, with the existing methods and their limitations. By outlining these challenges, we create the groundwork for presenting our proposed solution in Chapter 6.

Chapter 6: Coherent and Interpretable Forecasting Model This chapter details our second contribution, focusing on multi-step ahead forecasting and dynamic system discovery. We propose a novel framework that offers both accuracy and interpretability, overcoming the limitations of previous approaches.

Chapter 7: Conclusion and discussion This concluding chapter summarizes the thesis's key findings and contributions. We will also discuss potential limitations and future directions for research in this field.

1.5 Contributions

- KDD article: Causal and interpretable rules for time series analysis (Dhaou et al., 2021)
- Patent: procédé de contrôle d'un système et produit programme d'ordinateur associé. Amin Dhaou, Antoine Bertoncello, Sébastien Gourvénec, Josselin Garnier, Erwan Le Pennec. TotalEnergies OneTech, Centre National de la Recherche Scientifique, Ecole Polytechnique. France, N° de brevet: FR3124868B1, N° d'enregistrement national: 21 07171, 2023 ⟨hal-04455926⟩.
- Code: https://github.com/amindh/CAPP
- Article: Learning from mistakes: an Interpretable and Coherent Multi-step Ahead Time Series Forecasting Framework
- Code: https://github.com/amindh/multi_step

CHAPTER 2

Understanding ML models

Contents

This chapter presents an overview of eXplainable AI (XAI). Section 2.1 reviews the historical background and definition related to XAI. Section 2.2 establishes key definitions for interpretability, explainability, and causality in AI. Section 2.3 explores the established taxonomy within the literature. A concise review of the current state-of-the-art for XAI techniques is provided in Section 2.4. Section 2.5 focuses on the evaluation processes and methodologies used to assess interpretability and causality. Finally, Section 2.6 concludes the chapter by summarizing the main insights and highlighting the remaining challenges in the field.

2.1 Introduction

2.1.1 A brief overview of the history of Explainable AI

While early works related to model's explanation date back to the 1980s, when researchers showed interest in the explanation of expert systems (Moore & Swartout, 1988; Shortliffe & Buchanan, 1975), rule-based explanation on neural networks (Andrews et al., 1995) and recommendation systems (Herlocker et al., 2000), the formalization of this field with the term e**X**plainable **A**rtificial **I**ntelligence (XAI) was introduced by Van Lent et al. (2004). This concept aimed to "present the user with an easily understood chain of reasoning from the user's order" in a simulation gaming application for military objectives.

In recent years, AI research focused on developing highly accurate and efficient models, especially since the breakthrough of neural networks in 2012. While these models offer impressive efficiency and benefits across various sectors, concerns have arisen due to their opaque and complex nature, along with incidents of negative impacts (Telford, 2019; Yee et al., 2021; Završnik, 2020). This resulted in a shift within the field, urging researchers and developers to move beyond a single focus on performance and consider aspects like understanding, responsibility, and ethics. Indeed, the focus on XAI was renewed (Gunning, 2017), aiming to produce more explainable models while maintaining a high level of learning performance and enabling human users to understand, trust, and appropriately manage the model. It has been further developed and discussed to the present day, with different topics such as Interpretability, Explainability, and Causality.

Interestingly, Rudin (2019) (Rudin et al., 2022) argue that the perceived trade-off between XAI and model accuracy may not be as marked as it seems (i.e., "there is no scientific evidence") and suggest that efforts to understand better how systems arrive at their decisions can lead to improved performance and reliability, fostering trust and acceptance in these powerful tools. This ongoing discourse highlights the need for a nuanced approach that balances the advantages of AI with explanations. All these concerns have led to the emergence of research on new topics and algorithms since 2014 (Gunning, 2017), such as explaining black box models (Zeiler & Fergus, 2014) or models' decision (Ribeiro et al., 2016) and developing more transparent models (Letham et al., 2015).

Although there is no unanimous consent over the true objective and definition of XAI, several questions, both general and subjective, have been raised (Gunning, 2017; Lipton, 2017), including:

- ❖ Why did the model do that? Why not something else?

- ❖ When does the model succeed? When does it fail?
- ❖ What are the conditions to trust a model?
- ❖ Will it work once deployed?
- ❖ What else can it tell about the world?

Regulations, such as the General Data Protection Regulation (GDPR) (Radley-Gardner et al., 2016), underscore the importance of explanation in automated decision-making affecting individuals. Indeed, it is a considerable effort that strengthens data subject rights by introducing the *right to an explanation of algorithmic decision* and the *right to be informed* (article 22, 13), conditioning the organization using automated decision-making to clarify their underlying system to be challenged if needed. Programs have been developed in multiple countries in order to produce understandable models to attain objectives such as accountability, trust, and fairness ({France AI Strategy Report}, 2018; Gunning & Aha, 2019; {Royal Society}, 2017). Some researchers and companies tried to raise awareness and strike a balance between technological progress and ethical responsibility, and a thousand of *AI Giants* even asked for a pause or to slow down in advancement {Future of Life Institute}. For over three years, intense discussions have been taking place to establish a European law on AI, the AI Act ({European Commission}, 2021). On December 9, 2023, a political agreement was reached to regulate and legislate on the trustworthiness of AI {European Parliament} (2023). This achievement marks a significant step forward in ensuring that AI is developed and used responsibly and ethically.

Despite the growing importance of XAI, there is a lack of consensus on the precise definitions and objectives. Researchers from different fields of study proposed various definitions and taxonomies, leading to multiple approaches and methodologies shaped by individual perspectives and goals. Without a unified framework, accurately assessing and comparing models becomes a complex task, potentially hindering progress in XAI development (Nguyen & Martínez, 2020).

The diverse perspectives on XAI necessitate a clear presentation of the challenges, the current state of the field, and the definitions to clarify the scope of our work. After briefly introducing the epistemological definitions of causation, explanation, and interpretation, we describe the definitions of causality, explainability, and interpretability used in Machine Learning. Then, we present the taxonomy used in the state of the art and delve deeper into the techniques employed, focusing on interpretable and causal models. Finally, we describe how to evaluate XAI models.

2.1.2 The Challenge of Explainable AI

The rise of Explainable AI has attracted much interest as a means to enhance understanding of complex Machine Learning models. While pursuing XAI is commendable, it is crucial to recognize this domain's inherent complexities and multifaceted nature.

First, due to the complexity of real-world problems with large amounts of multivariate data with non-linear patterns, many complex and "opaque" ML models have been developed to capture these dynamics. They exhibit limitations in their design with the inherent algorithmic complexity that makes them called "black box" or on the technical aspect with inaccurate decisions, outlier predictions, and a lack of generalization beyond the training data. Such

limitations may necessitate a deeper understanding of the model's inner mechanisms to identify potential sources of error and refine the decision-making process.

XAI Goals There are many reasons behind the search for an explanation that needs to be investigated in order to respond in the best possible way. Indeed, XAI aims to build models that are trustworthy, informative, fair, and ethical. This ensures that they are reliable, safe, robust, and respect privacy while also providing insight into their decision-making process, offering confidence to the user (Barredo Arrieta et al., 2020; Carvalho et al., 2019; Doshi-Velez & Kim, 2017; Guidotti et al., 2019; Lipton, 2017). In this regard, the question of how to achieve these desiderata is a central tenet of XAI and is a challenge as they are subjective and depend on many factors, making them difficult to define and measure. For instance, trust is frequently cited as a key objective in XAI. However, its definition remains subjective. While some link trust to the model's confidence, understood as a model's ability to produce accurate predictions on the training data, Lipton (2017); Rudin et al. (2022) argue for a more nuanced understanding. Lipton (2017) asserts that trust arises not solely from "how often a model is right," but also from discerning "for which examples it is right". This distinction emphasizes that a model exhibiting high confidence during training may fail in deployment due to inherent biases, robustness limitations, or ethical concerns. Besides, Rudin et al. (2022) further emphasizes that XAI does not guarantee trust, but rather empowers users with the information necessary to assess the model's trustworthiness based on their own criteria.

User-dependent The inherent subjectivity of XAI can be attributed, in part, to the diverse range of stakeholders targeted by its various techniques. The level of understanding and nature of the explanation required will depend on the specific goals, background knowledge, and expertise of each stakeholder group. Existing literature identifies several distinct categories (Barredo Arrieta et al., 2020; Bhatt et al., 2020; Liao & Varshney, 2022; Preece et al., 2018):

- **Model developers/ Data Scientist**: This group primarily seeks to enhance the model's efficiency.
- **Business leaders (managers, executive board members)**: Their primary concerns lie in assessing compliance with regulations and comprehending the model's utility in achieving their business objectives.
- **Domain experts/ Model Users**: This group requires trust in the model's decision-making process and necessitates informed insights to guide their subsequent actions.
- **Impacted Users**: This group seeks to understand the rationale behind the model's decisions and assess their fairness and legitimacy.
- **Regulatory entities**: Their primary function is to audit and certify the model's compliance to legal and ethical standards regarding, for instance, privacy and safety concerns.

Domain-dependent Another contribution of the subjective nature of XAI is due to the domain of application in which it is implemented, raising specific challenges and considerations. Indeed, expectations and requirements for XAI vary significantly across domains. In high-risk applications like healthcare (Chaddad et al., 2023), aviation (Degas et al., 2022), military

(Griffin et al., 2022), and nuclear energy (Ayodeji et al., 2022), where human safety and lives are at stake, XAI is crucial. These domains demand a thorough understanding of the model's decisions to ensure they align with ethical considerations, mitigate biases, and maintain robustness in real-world scenarios. In these sensitive domains, the benchmark for comparison often resides in human experts' decision-making capabilities or traditional methods in the field.

The pursuit of XAI is a complex and multifaceted endeavor, encompassing diverse perspectives and methodologies. To fully grasp the nuances of XAI, it is essential to first delve into the epistemological underpinnings of causation, explanation, and interpretation. These concepts provide a foundation for understanding the nature of knowledge and how it informs our comprehension of ML models. In the next section, we delve into these concepts.

2.1.3 Epistemological Definitions

This section deals with the notion of "understanding", which is the mental process of grasping and making sense of information, a concept or a phenomenon. We explore how this intricate process is articulated and expressed through causation, explanation, and interpretation.

2.1.3.1 Causality

Definition 2.1.1 (Causality (Cambridge Dictionary))**.** *The principle that there is a cause for everything that happens.*

Causation refers to the relationship between cause and effect, where an event or phenomenon (cause) initiates or contributes to the occurrence of another event or phenomenon (effect). The study of causality has been a subject of intense philosophical debate for centuries, with various perspectives on the nature and identification of causal relationships.

The objective is not to make a survey on causality, but to give some main theories that have been developed. Many theories in causality can be related to counterfactuals, aiming to answer the question: "What would have happened if" (Miller, 2018). Among them, we can cite:

- **Regularity theories**: Aristotle's Regularity Theory suggests that events are linked as a regular sequence of causes and effects. He argues that there must be a necessary and sufficient condition for an event to occur (Falcon, 2023). Hume (1894) argues that causation is a regularity with empirical and repeated observation of a pattern. Hence, causality can be inferred only based on repeated observations of regularities, such as the occurrence of one event after another. Indeed, "if the first object had not been, the second never had existed" (Hume, 1894).
- **Possible worlds**: Lewis (1973) argues that causation is determined by comparing actual events to hypothetical "possible worlds" where things would have happened differently. The most similar world, where a minimal change leads to a different outcome, represents the true cause. However, this approach relies on subjective judgments of "similarity", leading to a "miracle" scenario (Pearl, 2013) and raising concerns about subjectivity and reliability.
- **Process theories**: These theories assume the existence of a physical or metaphysical process driving causation. The emphasis lies on identifying the mechanism – such as

energy transfer – that connects cause and effect, providing a deeper understanding of their interaction. Indeed, Salmon (1994)'s theory suggests causation happens through continuous processes instead of discrete events.

- **Interventionist theories**: Pearl & Halpern (2005) suggest that if intervening on a potential cause results in a change in the expected effect, then causality can be established. This theory often connects to probabilistic frameworks, defining causality as the ability to increase the probability of an effect through cause's manipulation.

Although numerous theories address causality, Pearl & Mackenzie (2018) proposed a unifying framework with three distinct levels of causality: 1) the **association** level focuses on identifying correlations and patterns between variables, 2) the **intervention** level actively manipulates a potential cause and observes the effect's response, 3) the **counterfactual** level involves imagining alternative realities where the cause is absent and analyzing hypothetical effects.

Multiple challenges exist in this domain, including identifying causality when there are unobserved variables, multiple necessary causes (set of events all necessary to cause an event), or multiple sufficient causes (multiple possible ways to cause the event where only one is required.

2.1.3.2 Explanation and Interpretation

In philosophy and science, both explanation and interpretation seek to justify findings, but also to deepen understanding and even persuade others of their validity.

Definition 2.1.2 (Explanation (Cambridge Dictionary))**.** *The details or other information that someone gives to make something clear or easy to understand.*

Definition 2.1.3 (Explain (Merriam-Webster))**.** *To make plain or understandable; to give the reason for or cause of; to show the logical development or relationships of.*

Definition 2.1.4 (Interpretation (Cambridge Dictionary))**.** *An explanation or opinion of what something means.*

Definition 2.1.5 (Interpret (Merriam-Webster))**.** *To explain or tell the meaning of; present in understandable terms.*

In the literature, explanation and interpretation are often used interchangeably. Both seek understanding, but with different angles: explanation delves into the "why" through logic or cause-and-effect, while interpretation focuses on the "what" by uncovering meaning and context. Though "interpretation" has a more subjective connotation, its goal aligns with explanation: to uncover the truth behind phenomena or events. In this discussion, we consider them both as explanations for simplification.

Explanation is a central concept in understanding phenomena in different domains, aiming to provide answers to questions such as why, what, or how. Throughout history, scholars have offered diverse theories attempting to capture the essence of explanation. We describe some key concepts in the literature (Mayes, 2001; Miller, 2018; Srinivasan & Chander, 2021):

- **Deductive-Nomological model**: One of the classical types of explanation were proposed by Hempel & Oppenheim (1948) in the form of logical proofs. Starting from assumptions

with general laws and initial conditions, deductive arguments lead to the phenomenon to be explained.

- **Causal Patterns**: Several prominent theories emphasize the importance of causality in explanation. (Glennan, 1996) defines uncovering the causal mechanisms governing a phenomenon as the ultimate explanatory goal. This aligns with Halpern & Pearl (2005)'s view that an explanation is essentially a potential cause for the observed phenomenon, regardless of initial uncertainty. Lewis (1986) argues that an explanation provides the event's causal history, while Josephson & Josephson (1996) defines explanation as assigning causal responsibility.

- **Mental models**: Bridging traditional AI and neuroscience, mental models (Holland, 1986) are internal maps built from rules connecting situations and actions. They act as simplified internal representations of the world, built from interconnected "if-then" rules. If the initial understanding fails, then the brain searches deeper levels of this map to understand why. In this theory, explaining involves adjusting these rule-based maps until they fit what we experience.

Other approaches explain through different perspectives (Srinivasan & Chander, 2021), such as **mechanical** explanations with physical objects and interactions, **illustrations** with examples, comparisons, analogies and counterfactuals and finally **intentional** explanation which concern beliefs and desires.

Halpern & Pearl (2005) argued that our "epistemic state", or state of knowledge, fundamentally shapes what we accept as an explanation. Indeed, what constitutes a satisfactory explanation depends on what we already know. Even "good" explanations, if accurate, may not always be easily grasped. The multifaceted nature of explanation can be explored through various lenses (Angelopoulou et al., 2022):

- Social attribution: understanding how people explain behavior by analyzing intentionality, beliefs, desires, and intentions.

- Cognitive processes: delving into the cognitive biases, norms, and changing nature of explanations and how we evaluate them, including counterfactual reasoning.

- Social explanation: exploring how explanations are communicated through spoken or signed language, dialogues, and even non-linguistic means.

2.2 Explainability, Interpretability in ML

In the area of AI and Machine Learning, eXplainable AI (XAI) is a concept that encompasses all types of explanations, from the analysis of the relationships in the data to the core structure and decision-making process of a machine learning model. The aim of this development is to make all the models understood, to some extent, by giving the ability to understand the internal mechanisms, underlying theory, and decision-making process that lead to an outcome. It also focuses on communicating the decision in an understandable way, with meaningful and justified explanations. This area rose in parallel with the growing number of "black box" models and aims to cope with their complexity. The development of this domain has shown to be

necessary due to the advent of regulations and to ensure trust to users. Indeed, one of the goals of XAI is to be persuasive and informative from the model's conception to its use.

2.2.1 Challenges in defining the concepts

Researchers in the XAI domain use terms including causality, interpretability, and explainability and aim to describe the ML models and algorithms. While the concept of causality carries the same meaning within both XAI and traditional philosophy seen in part 2.1.3.1, we do not delve into its specific definition here. However, we found in the literature that the concepts of interpretability and explainability are often used interchangeably without any difference (Du et al., 2019; Guidotti et al., 2019; Lipton, 2017; Miller, 2018; Mittelstadt et al., 2019; Molnar, 2020; Murdoch et al., 2019) and from articles that differentiate them (Doshi-Velez & Kim, 2017; Montavon et al., 2018), we can find opposite definitions. A difference with philosophical definitions stands in ML as it takes a more pragmatic role by trying to understand the model or the decision-making process of an algorithm.

Despite significant efforts to establish a unified definition allowing to better design and evaluate AI systems (Zhong & Negre, 2021), achieving this goal remains challenging due to the inherent subjectivity and diverse research objectives (Marcinkevičs & Vogt, 2023; Miller, 2018). Rudin (2019) highlights another layer of complexity by arguing that a single, all-encompassing definition might be impractical and unnecessary, given the domain-specific nature of AI and the potential drawbacks of such a rigid approach (Rudin et al., 2022).

2.2.2 Definitions in the literature

The terms "interpretability" and "explainability" are often used interchangeably but with some differences. Authors like Kim et al. (2016) define interpretability based on the user's ability to predict the model's results, Caruana et al. (2015) define it based on the intelligibility independent of the user background, while Ribeiro et al. (2016) focuses on understanding the relationship between inputs and outputs. However, most authors delve deeper and explore various notions: Lipton (2017) differentiates between "transparent models," where humans can understand the decision-making process, and "post-hoc explanations," which explain the model's reasoning after the inference. Similarly, Gilpin et al. (2018) distinguishes between comprehending and describing the model's internal mechanism and the ability to describe and provide the causes of a neural network's decision. They highlight a trade-off between interpretability and completeness (accuracy of the explanation). Miller (2018), while not differentiating the terms themselves, defines interpretability (based on Biran & Cotton (2017) definition) as the degree to which the user understands the cause of a decision. He distinguishes this from "justification," which explains why a decision is good without delving into the process. Barredo Arrieta et al. (2020); Doshi-Velez & Kim (2017); Guidotti et al. (2019) aligns with this distinction, differentiating between explaining the model itself and explaining individual decisions without the full decision-making process.

A review of the literature reveals that two key concepts consistently emerge. The first centers around the ability to articulate a model's decision-making process in terms understandable to a user. The level of required "understandability" inherently constrains the choice of model, necessitating a trade-off between model complexity and understanding. The second concept focuses on the ability to provide accurate explanations of a model's decisions without delving

into the entire decision-making process. This approach aims to provide insightful justifications for the model's output without overwhelming the user with technical details. This entails offering explanations that are adapted to the user's level of understanding, ensuring effective communication of the model's reasoning.

2.2.3 Definitions in this thesis

Throughout this thesis, to ensure clarity and consistency, "interpretability" and "explainability" will be understood according to the following two established definitions:

Definition 2.2.1 (Interpretability)**.** *Interpretability is the ability to describe a model's decision-making process to a user. We call these models interpretable.*

Definition 2.2.2 (Explainability)**.** *Explainability is the ability to provide an explanation of the reason behind a decision of a model to a user. We call these models explainable.*

The effectiveness of XAI methods depends heavily on the application context. The domain, user's goals, and specific tasks all play a crucial role in determining the most suitable approach. In the next section, we present a taxonomy of XAI concepts based on (Carvalho et al., 2019; Molnar, 2020) work, aiming to guide the model's selection based on these different factors.

2.3 Taxonomy

XAI models can be explained at different levels of granularity, from explaining the data and the overall model's behavior to a single instance. The different levels provide insight into the data, how the model works, and how it makes decisions. In the following, we delve into the proposed taxonomy and describe the various levels existing in the literature.

2.3.1 Data or Model focused Explanation

Data-focused methods aim to provide insights into the data used to train the model. They can identify potential biases or limitations in the data that may affect the model's performance, such as selection bias and the presence of outliers. It aims, for instance, to reveal the relationships between features, extract patterns, and identify causal relationships as shown in section 2.4.3. This type of explanation can be crucial for ensuring data quality and preventing the model from making biased decisions. In the literature, this field is commonly referred to as "exploratory data analysis" or "data analysis" (Tukey, 1977). A multitude of methods is employed to unravel the complexities of data sets: statistical methods such as the mean, median or standard deviation provide valuable insights into data distribution and may facilitate missing value analysis. Correlation analysis, employing measures like Pearson or Spearman correlation, quantifies the strength of a relationship between variables. Additionally, statistical techniques like the IQR method and Z-score, along with model-based approaches such as Isolation Forest (Liu et al., 2008), Local Outlier Factor (Breunig et al., 2000), or visualization techniques like box plots, aims to identify and isolate outliers in data. Dimensionality reduction methods like Principal Component Analysis (Pearson, 1901) or t-SNE (Hinton & Roweis, 2002), aim to reduce the number of features while preserving essential information. Other areas, such as data

visualization tools, clustering, and time series analysis, are described in (Mukhiya & Ahmed, 2020; Wickham, 2016).

These methods encompass descriptive techniques that not only visualize but also summarize, transform, and analyze data.

Model-focused methods are designed in order to explain the model's inner workings and/or how it makes decisions and predictions. These models are usually divided into two categories:

- **Intrinsic**: Models that are interpretable or explainable by design.
- **Post-hoc**: Explanation methods that are applied after the model training (often black box model).

More nuanced aspects of these notions are examined in the following section.

2.3.2 Intrinsic or Post Hoc

2.3.2.1 Intrinsic methods

Intrinsic methods encompass two main categories:

- **Transparent models**: these models are designed or trained to be inherently transparent, such as linear regression or decision trees. Lipton (2017) introduce three types of transparency:
  - **Simulatability**: defines the ability of a human to mentally grasp a model's entire decision-making process, from input data and parameters to prediction in a reasonable time. While simple models like decision trees are generally simulatable, they can become complex as they grow larger and handle more complex data. Ultimately, what constitutes "reasonable" depends on the limitations of human cognition.
  - **Decomposability**: defines the ability to explain different parts of the model such as inputs, parameters, and calculation. For example, each node of the decision tree is explained by a simple rule or the parameters of linear regression represent association strength.
  - **Algorithmic transparency**: defines the ability to understand the learning algorithm. Algorithms such as linear regression are simulatable and decomposable but may be biased due to imbalance or extreme values data set, for example. This definition can be linked to fairness.
- **Intrinsic explainable models**: these models are designed to incorporate parts that explain the reasoning of the model. For instance, attention models involve attention weights that reveal the model's focus on input elements for its predictions, as shown in Lim (2018) article.

2.3.2.2 Post-Hoc methods

Post-hoc explanation methods are designed to explain a model after its training process. They generally do not provide a global understanding of decision-making processes. They can be applied to black-box models, and because they are not part of the model, they do not reduce predictive performance. A popular post-hoc explanation method is SHAP (SHapley Additive exPlanations) (Lundberg & Lee, 2017) which explains how much each feature contributes to the prediction by assigning them importance scores. A major drawback of these methods is that they may lead to discrepancies between the given explanation and what the models actually do.

The required level of understanding dictates the choice of model type. Transparent models are adapted for a precise understanding of the model's decision-making process, facilitating, for instance, the identification of performance limitations in the model's architecture or parameters and even helps to mitigate algorithmic bias. Conversely, for delving into specific predictions, post-hoc methods provide valuable insights. Ultimately, methods that comprehensively integrate data and model understanding may offer the most robust and reliable explanation.

2.3.3 Model Specific or Model Agnostic

Model Specific explainable methods deal with the unique characteristics and structure of a particular machine learning model or family of models. This approach focuses on the properties of the model to extract insights and explanations. For instance, feature importance analyzes the influence of individual features on the predictions of decision trees and ensemble methods like random forests (Breiman, 2001).

Model Agnostic methods can be applied to any type of architecture, independently of the internal structure of a specific machine learning model. It focuses on explaining the output without knowing the mechanism that produced it. For instance, SHAP estimation is solely based on model outputs. Another advantage of this approach lies in its ability to facilitate the comparison of performance across different models, establishing a common benchmark for evaluation.

2.3.4 Local or Global explanation

Explanations of ML models can be provided at different levels of granularity, ranging from local to global explanations.

Local explanation models aim to extract the reasons or decision-making process of a particular prediction made by a model. The local explanation would answer questions such as

- Why did the model make this specific prediction?
- Which feature contributed most to this instance?
- How confident is the model on this specific prediction?

For instance, SHAP provides a local explanation as it estimates the contributions of each feature for a specific instance or prediction.

Global explanation seeks to provide general and global insight of the model over the features and the whole data set. The global model would answer different questions such as

- How does the model behave on average?
- What are the most important features contributing to the overall model?
- What is the model decision boundary?

For instance, the decision tree's feature importance offers global explanations by aggregating the impact of features across all observations.

The difference in granularity or type of explanation model is important as it is often required by the different stakeholders that have different needs. The following section explores key methods and concepts in interpretability and causality, reviewing prevalent approaches. It also briefly introduces explainability concepts and some of the main approaches in this field.

2.4 State of the art

2.4.1 Explainability

In the context of an increasing number of complex models, explainability plays a role by clarifying and articulating the reasoning behind the model's decision. Doran et al. (2017) describe these models as comprehensible with a capacity to provide insight to the users on how a conclusion is reached. For instance, Lipton (2017) describes human decision-makers as explainable as they convey useful information from the brain, which is a black-box model that operates through a mechanism of thinking by gathering and processing data in order to extract a decision. In the realm of ML and real-world applications, it allows adding reasons, evidence for decisions, and prediction by showcasing the factors that influenced the outcome.

Various methods have been proposed in the literature to address this challenge. In the literature, the methods are on different levels: 1) Intrinsic explainable models such as attention models (Lim et al., 2021) 2) Post-hoc models that are either a) model specific such as random forest feature importance (Breiman et al., 1984) or b) model agnostic such as SHAP. In addition, the methods can be local, describing a specific instance, or global.

To gain a comprehensive understanding of the major explainability models, we shall employ Barredo Arrieta et al. (2020) categorization to describe some of the prominent methods, while directing the reader to comprehensive surveys for a more in-depth exploration of this field (Barredo Arrieta et al., 2020; Chaddad et al., 2023; Guidotti et al., 2019; Linardatos et al., 2021; Marcinkevičs & Vogt, 2023). In each level of granularity, methods can be of different types:

- **Explanation by simplification** comprise methods that provide local explanations such as LIME (Local Interpretable Model-Agnostic Explanations) (Ribeiro et al., 2016), which explains individual predictions by locally approximating the model around the data point, and distillation methods that trains an interpretable model (e.g., linear models (Tan et al., 2018), rule-based, decision tree (He et al., 2020)) to mimic the complex model's behavior.

- **Feature relevance explanation** methods identify and rank features based on their contributions to the model's prediction such as sensitivity analysis with SHAP or random forest feature importance.
- **Visual explanation** methods use visualizations to explain the model's behavior, such as Grad-CAM (Gradient-weighted Class Activation Mapping) (Selvaraju et al., 2017) which uses gradients to highlight image regions crucial for a deep learning model's prediction.
- **Explanation by example** methods aim to provide similar data instances to explain the reasons behind a specific prediction. It can be in the form of example-based explanation, such as in (Kim et al., 2016), which captures both good examples that are "well classified" (prototypes) and challenging instances that are "misclassified" (criticisms) from the data and counterfactual explanations that simulate what-if scenarios to provide similar output by perturbing specific input features (Mothilal et al., 2020).
- **Text explanation** methods generate natural language descriptions to explain model predictions, such as image's caption generation (Xu et al., 2015).

While not explicitly addressed here, it is important to note that some techniques are specifically designed for deep learning models due to their inherent complexity. The literature offers diverse explainable methods, which implies that the proposed categorization is neither mutually exclusive nor definitive, as some methods may exhibit characteristics that place them in multiple categories.

2.4.2 Interpretability

While explainability focuses on providing explanations for model decisions, interpretability explores in greater depth the transparency and understanding of the model's inner workings. It emphasizes the degree to which a human can comprehend the decision-making process employed by the model (Doshi-Velez & Kim, 2017; Molnar, 2020). In this context, the goal extends beyond simply explaining outcomes by establishing the relationships between inputs and outputs. This is usually achieved using simpler models where such relationships can be readily identified. This simplicity enables us to trace the causal pathways of the algorithm and identify the interaction of features that contribute to the model's decisions. However, this simplicity often comes at the price of a concession to accuracy, presenting a challenge for real-world applications that demand both interpretability and predictive power (Caruana et al., 2015; Wang et al., 2015; Wang, 2019).

Interpretability can be viewed on multiple levels, ranging from comprehending the overall model architecture and its constituent components to understanding the specific decision-making process for individual predictions. This deeper understanding allows one to identify potential biases, assess the model's robustness, and make informed decisions about its deployment and use.

A multitude of methodological approaches have been proposed in the literature to address this challenge. We delve into a selection of these methods in the subsequent section while directing readers to comprehensive reviews for a more in-depth exploration of this domain (Barredo Arrieta et al., 2020; Burkart & Huber, 2021; Marcinkevičs & Vogt, 2023; Rudin et al., 2022).

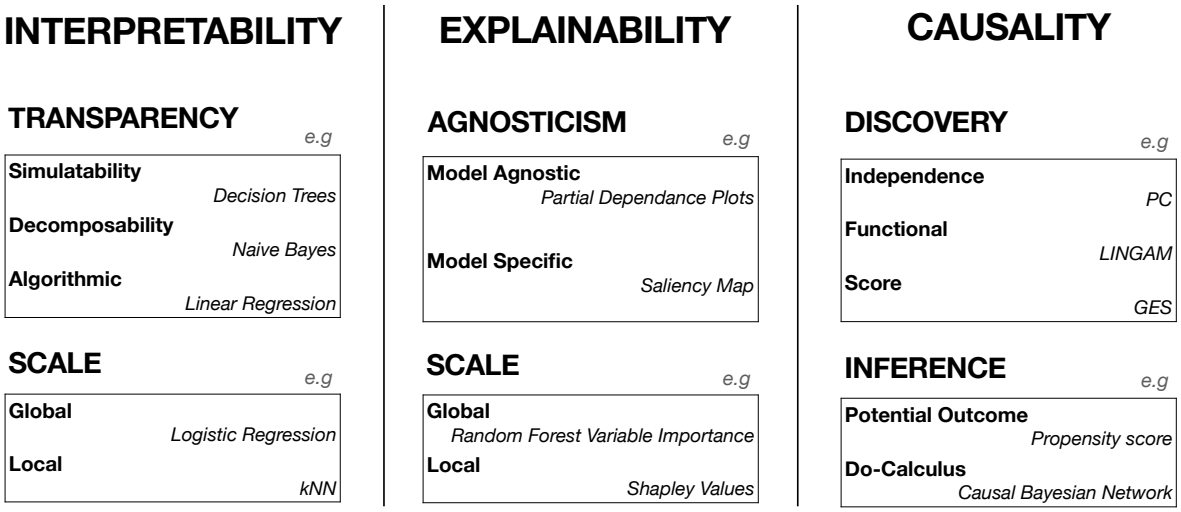


Figure 2.1: This figure represents the main categories of the interpretability, explicability and causality domain (adapted from (Marcinkevičs & Vogt, 2023)). Examples of models are shown in italics.

To effectively describe the various methods, it is essential to establish the notation. Let us introduce a random vector $\mathbf{X} = (X^{(1)}, \ldots, X^{(p)})$ of dimension $p \in \mathbb{N}$ where for each $k \in \{1, \ldots, p\}$, $X^{(k)}$ takes its values in $\mathbb{R}$ and a random variable Y taking its values in a domain $\mathcal{Y} \subseteq \mathbb{R}$, which can be a finite or a continuous set for classification and regression problems respectively. The training data set is represented as $\mathcal{D}_N = \{(\mathbf{x}_i, y_i) | i = 1, \ldots, N\}$, where each $(\mathbf{x}_i, y_i)$ pair is drawn from the joint distribution of $\mathbf{X}$ and Y and where $\mathbf{x}_i = (x_i^{(1)}, \ldots, x_i^{(p)})$. Additionally, let $\mathbb{X} = (\mathbf{x}_1^\intercal, \ldots, \mathbf{x}_N^\intercal) \in \mathbb{R}^{N \times p}$ be the design matrix and $\mathbf{y} = (y_1, \ldots, y_N) \in \mathcal{Y}^N$ be the target vector. In the following, for simplicity, we refer to $(\mathbf{x}, y)$ for a single instance where $\mathbf{x} = (x^{(1)}, \ldots, x^{(p)})$.

2.4.2.1 Regression

Linear regression assumes a linear dependency between predictor variables (inputs) and continuous target variables (outputs), it is expressed as a linear equation of the form:

$$Y = \beta_0 + \beta_1 X^{(1)} + \beta_2 X^{(2)} + \ldots + \beta_p X^{(p)} + \epsilon \tag{2.1}$$

where $\beta_0 \in \mathbb{R}$ is the intercept, $\beta_k \in \mathbb{R}$ the coefficient for $k \in \{1, \ldots, p\}$ and $\epsilon \sim \mathcal{N}(0, \sigma^2)$ is the error term with constant variance σ^2. A common approach to estimate the parameters $\beta = (\beta_0, \ldots, \beta_p)$ involves minimizing the sum of squared errors, also known as the least squares objective function. The objective function is expressed as follows:

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}} \, \|\mathbf{y} - \mathbb{X}\beta\|_2^2 \tag{2.2}$$

where $\| \cdot \|_2$ is the ℓ_2 norm. Linear regression is interpretable by design as the target in the model is linearly related to each of the variable $X^{(k)}$ by a parameter β_k that gives a measure of its influence. Its validity relies on certain assumptions, such as the linearity of the underlying

relation, the absence of multicollinearity (interdependent predictors), Gaussian noise (normally distributed residuals), and homoscedasticity (constant variance of residuals). However, linear regression suffers from drawbacks such as the sensitivity to outliers, lack of sparsity and lack of flexibility as it cannot capture non-linear relationships.

Penalized Regression Penalized/Regularized regression methods are particularly useful to address these limitations. The objective function is defined as :

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}} \frac{1}{N} \|\mathbf{y} - \mathbb{X}\beta\|_2^2 + \lambda R(\beta) \tag{2.3}$$

where $R(\beta)$ is the regularization function and $\lambda \geq 0$ is the regularization strength. The main regularization functions include:

- $R(\beta) = \|\beta\|_0$, named ℓ_0 regularization, penalizes the number of non-zero coefficients, promoting sparsity. This penalty exhibits limitations in practice due to its non-convexity and combinatorial nature.
- $R(\beta) = \|\beta\|_1$, named ℓ_1 regularization, penalizes the absolute value of the coefficients (penalize large coefficient), promoting sparsity.
- $R(\beta) = \|\beta\|_2^2$, named ℓ_2 regularization, penalizes the square values of the coefficients, controlling the magnitude of the coefficient.

These regularization methods can be combined to form more sophisticated regularization techniques. One popular example is Elastic Net (Hastie & Zou, 2005), which combines both ℓ_1 and ℓ_2 regularizations. This approach offers the benefits of both types of regularization and is defined as:

$$R(\beta) = \alpha \|\beta\|_1 + (1 - \alpha) \|\beta\|_2^2 \tag{2.4}$$

where $\alpha \in [0, 1]$.

Other methods exist, including Partial Least Square (PLS) regression (Wold et al., 2001), which identifies the most relevant variables in a data set by summarizing the relationships between variables through latent variables and Group Lasso regularization (Yuan & Lin, 2006) which encourages sparsity within groups of variables, leveraging domain knowledge to define these groups.

Logistic regression focuses on predicting the probability of a binary target variable $\pi = P(Y = 1|\mathbf{X} = \mathbf{x})$ based on input features. The main assumption of the model is that we can approximate this probability through the sigmoid (logistic) function $g : \mathbb{R} \to [0, 1]$ defined as :

$$g(z) = \frac{1}{1 + exp(-z)}. \tag{2.5}$$

and which can be expressed with the logit function

$$g^{-1}(\pi) = logit(\pi) = \ln \frac{\pi}{1 - \pi}.$$

The relation between the predictor variables and the probability of the outcome is then expressed as:

$$P(Y = 1|\mathbf{X} = \mathbf{x}) = g(\beta_0 + \beta_1 x^{(1)} + \beta_2 x^{(2)} + \ldots + \beta_p x^{(p)})$$

The estimation of the coefficients β is done by maximizing the likelihood function, which is equivalent to minimizing the cross-entropy loss function.

Generalized linear models (GLMs) (Nelder & Wedderburn, 1972) are an extension of linear regression model allowing for various response distributions beyond the Gaussian noise assumption. GLM consists of three components: 1) a distribution for modeling $Y|\mathbf{X} = \mathbf{x}$, 2) a relation between the parameter and the predictor $\mathbf{x}\beta$ 3) A link function g that connects these two components. The link function can be chosen based on the characteristics of the target variable. The relation is expressed as:

$$E(Y|\mathbf{X} = \mathbf{x}) = g(\beta_0 + \beta_1 x^{(1)} + \beta_2 x^{(2)} + \cdots + \beta_p x^{(p)}) \tag{2.6}$$

In logistic regression, the link function g is the sigmoid function, mapping the weighted sum of features to probabilities between 0 and 1. Similarly, the estimation is done by maximum likelihood.

Due to linear models' limitations in capturing non-linear relationships, other techniques have been introduced to improve the flexibility of the models.

Non-linear models

- **Feature transformation** extends the applicability of linear models by applying non-linear functions to the input features, expanding the model's ability to capture more complex patterns. One common approach is polynomial expansion, which generates additional variables based on polynomials and higher-order interactions between the original features.
- **Generalized additive models (GAMs)** extend linear models and particularly GLMs by capturing non-linear relationships between the input and the target variable. It models the response as a sum of smooth, non-parametric functions f_j for $j \in \{1, \ldots, p\}$ for each feature as follows:

  $$E(Y|\mathbf{X} = \mathbf{x}) = g\left(\beta_0 + f_1(x^{(1)}) + f_2(x^{(2)}) + \cdots + f_p(x^{(p)})\right) \tag{2.7}$$

  While less inherently interpretable than linear models, GAMs excel at uncovering complex patterns in diverse data scenarios.

2.4.2.2 Rule-based models

Rule-based models are one of the most popular models in machine learning and are known for their high interpretability. These models learn explicit sets of rules that capture patterns in the data and describe the decision-making process in a clear and understandable way. In this part, we describe two main approaches: decision rules and decision trees. We also explore some hybrid methods that combine these methods with other machine learning algorithms.

Decision Rules

Decision rules aim to uncover interpretable patterns within complex data. The rules built on specific feature values follow a general structure: IF the conditions are met, THEN make a certain prediction. Prediction can be made using a single or multiple rules. **A**ssociation **R**ule **M**ining (ARM) is a data mining framework that extracts rules from a database. It has the advantage of being highly interpretable and easy to understand. This part explores the origin of ARM in the retail domain and describes its utility in uncovering relationships.

Motivation **F**requent **I**temset **M**ining (FIM) has been developed by Agrawal et al. (1993) in order to discover interesting patterns, relationships, and associations in large data sets. One of the earliest and well known applications is in market basket analysis. Commercial enterprises accumulate a significant amount of data on a daily basis holding the key to understanding consumer behavior and driving effective business strategies. In this setting, FIM involves a database $\mathcal{D}_b$ comprising **transactions** $\mathcal{T}$, where each transaction represents a set of **items** I purchased by a customer from a set $\mathcal{I}$. For instance, as presented in table 2.1, $\mathcal{T} = \{\mathrm{T}_1, \mathrm{T}_2, \mathrm{T}_3, \mathrm{T}_4, \mathrm{T}_5\}$ and $\mathcal{I} = \{$milk, bread, butter, cheese, diapers$\}$. Note that items, represented as categorical variables, act as the input features for the association rule mining algorithm. FIM aims to discover subsets of $\mathcal{I}$ that are frequently purchased together.

Transaction ID	items
T_1	milk, bread
T_2	butter
T_3	cheese, diapers
T_4	milk, bread, butter
T_5	bread

Table 2.1: Example of a database containing five transactions, each associated with a set of items

The analysis of this type of data aims to uncover associations or relationships that would be challenging to identify manually, particularly for data sets containing the purchasing behaviors of millions of consumers. These relationships can be effectively represented in the form of association rules, such as the rule $\{\text{milk}, \text{bread}\} \implies \{\text{butter}\}$ extracted from Table 2.1.

Association rules are typically generated in two steps (Agrawal et al., 1993):

1. discovering frequent subsets of items in the database
2. generating rules using the frequent subsets

In the following, we outline the fundamental concepts and algorithms underpinning FIM. We begin by defining essential terminology and describing the primary algorithms used in FIM. Subsequently, we delve into the Apriori algorithm (Agrawal et al., 1993), the first FIM technique, and illustrate its application in a practical scenario.

Frequent Itemset Mining Framework The exponential growth of candidate rules as the database size increases poses a computational challenge. Considering a database containing N

items, the total number of item combinations grows as the sum $\sum_{k=1}^{N} \frac{N!}{(N-k)!}$. This factorial complexity renders the problem computationally intractable for large databases. Therefore, techniques to filter and prioritize relevant rules are crucial for extracting meaningful insights. We begin by defining the key concepts before describing the algorithm.

Definition 2.4.1 (Itemset and k-itemset)**.** *An itemset is a set of items and a k-itemset is a set of k items for $k \in \mathbb{N}$.*
An itemset I' is a subset of an itemset I if $I' \subset I$.
An itemset I' is a superset of an itemset I if $I \subset I'$.

For example, $\{\text{milk}, \text{bread}\}$ in table 2.1 is a 2-itemset. In the following, let I be an itemset of $\mathcal{I}$ from the database $\mathcal{D}_b$.

Definition 2.4.2 (Support)**.** *The support of an itemset $I \subseteq \mathcal{I}$ is defined as*

$$support(I) = \frac{|\{T \in \mathcal{T} | I \subseteq T\}|}{|\mathcal{T}|} \tag{2.8}$$

Remark 2.4.3. *The **support**, also called relative support, ranges within $[0, 1]$ and is the frequency of apparition of I within the transactions of the database $\mathcal{D}_b$. Some authors define it without normalization (Fournier-Viger et al., 2017).*

Definition 2.4.4 (Frequent Itemset and Minimum Support Count)**.** *An itemset $I \subseteq \mathcal{I}$ is a frequent itemset if, and only if, support(I) $\geq c$ where $c \in [0, 1]$ is the minimum support count.*

In other words, c is a threshold that determines at which frequency an itemset is considered frequent in the database. For clarity reasons, we denote "supcount" for the support count c in the following.

Frequent Itemset Mining Algorithm Numerous algorithms have been developed for extracting frequent patterns from databases. In the following discussion, we focus on the most prominent algorithms within specific categories (given in Fournier-Viger et al. (2017), providing a concise overview of their principles and applications. There are two main categories:

- Breath-first search algorithms: The algorithms such as Apriori involve exploring frequent itemsets of increasing size. They start by identifying individual items that appear frequently, i.e., frequent 1-itemset. Then, they build on these frequent 1-itemset to discover pairs of items that appear together frequently, i.e., 2-itemsets. This process continues, iteratively searching for larger groups of itemsets that frequently co-occur in the data.
- Depth-first search algorithms: Algorithms such as Eclat (Zaki, 2000), FP-Growth (Han et al., 2004), H-mine (Pei et al., 2001) and LCM (Uno et al., 2004) typically involve recursively growing itemsets by adding one item at a time (depth) and exploring the database to find frequent itemsets.

Apriori Algorithm This part introduces the Apriori algorithm (Agrawal et al., 1993), a widely adopted technique for mining frequent subsets, which we employ in chapter 4. To gain a deeper understanding of the Apriori algorithm's inner workings, we present a practical example and demonstrate its application.

The Apriori algorithm solves the computational issue and is based on two main principles also called anti-monotonicity:

1. Any non-empty subset of a frequent itemset is frequent i.e for a minimum support $s \in \mathbb{N}, \forall I, I' \subseteq \mathcal{I}$ such that $I' \subseteq I$ and $I' \neq \emptyset$, $\text{support}(I) \geq s \implies \text{support}(I') \geq s$.

2. Any superset of a non-frequent itemset is non-frequent i.e for a minimum support $s \in \mathbb{N}, \forall I, I' \subseteq \mathcal{I}$ such that $I \subset I'$, $\text{support}(I) < s \implies \text{support}(I') < s$.

These principles allow the algorithm to efficiently identify frequent itemsets by pruning the search space.

We present a practical example to demonstrate the application of the Apriori algorithm. Consider a data set consisting of 8 transactions involving 5 items and apply the Apriori algorithm. Note that the transactions represent itemsets found within the customer database.

T ID	items
T_1	i_1, i_2, i_5
T_2	i_1, i_4
T_3	i_1, i_2
T_4	i_1, i_2, i_3
T_5	i_2, i_3
T_6	i_2, i_3
T_7	i_2, i_3, i_5
T_8	i_1, i_2, i_5

The first step is to compute the support count of each item in the data set, which will be named C_1: the candidate set. We filter this candidate set with a minimum support count $supcount = 2$ which yields the itemset L_1 in table 2.2.

Candidate set C_1	supcount	itemset L_1
i_1	5	✓
i_2	6	✓
i_3	4	✓
i_4	1	✗
i_5	3	✓

Table 2.2: Candidate set C_1 and generated itemset L_1

The second step generates the candidate set C_2 from L_1. The itemset L_1 is joined with itself. For instance, considering the first item i_1, joining the table 2.2 with itself will create the itemsets $\{i_1, i_2\}, \{i_1, i_3\}, \{i_1, i_5\}$. All combinations between frequent items of L_1 follow the same principle to generate C_2.

Similarly to step 1, the candidates C_2 are then sorted based on the minimum support count in order to keep the frequent itemsets. This yields:

Candidate set C_2	supcount	itemset L_2
$\{i_1, i_2\}$	4	✓
$\{i_1, i_3\}$	1	✗
$\{i_1, i_5\}$	2	✓
$\{i_2, i_3\}$	4	✓
$\{i_2, i_5\}$	3	✓
$\{i_3, i_5\}$	1	✗

Table 2.3: Candidate set C_2 and generate itemset L_2

More generally, **joining L_k with itself** will create itemsets of size $k+1$ by combining two itemset of L_k where $k-1$ items called "keys" are equals. For example, in table 2.3, the self-join will create the itemset $\{i_1, i_2, i_5\}$ from the first two itemsets $\{11, 12\}$ $\{11, 15\}$ where i_1 is the "key".
This process outputs associations like $\{i_1, i_2\}$. In order to uncover associations with more than two explanatory items, we repeat the same process and compute recursively the candidates and the itemsets.

Association rule mining offers a valuable tool for uncovering relationships within large data sets. It efficiently computes and presents relationships as simple and interpretable rules, making them accessible to diverse audiences. However, it is important to acknowledge that as the dimensionality increases, the number of potential associations grows significantly, potentially leading to the identification of spurious associations. Therefore, careful selection and evaluation of the extracted rules are crucial.

In the next part, we explore decision trees, another decision rule approach that leverages a sequence of rules to build a predictive model.

Decision Trees Decision trees, like decision rules, uncover interpretable patterns within complex data and can be used for prediction for both classification and regression. Decision trees not only make predictions but also offer a visual representation of the decision-making process and a clear understanding of how each feature contributes to the final outcome.

Main concepts Decision trees employ a tree structure in order to make decisions by learning from input data using a set of if-else conditions. Starting from the root, each *branch* leads to a *node* where a *splitting criterion* is used on the input variable until a terminal node, called *leaf* representing the prediction, is reached. Other parameters characterize the tree, such as its size, corresponding to the number of nodes; the node depth, corresponding to the distance to the root; and the tree height, which is the depth of the lowest leaf.

Let us consider a supervised learning problem where each training sample vector $\mathbf{x}_i$, for $i \in \{1, \ldots, N\}$, is associated with a target y_i. For regression problem, $y_i \in \mathbb{R}$, while in classification problems $y_i \in \{1, \ldots, K\}$ where $K \in \mathbb{N}$ represents the number of classes. In each node V of the decision tree, a test is done on one of the features at a specific value $t_V^{(k)}$ for $k \in \{1, \ldots, p\}$. If the variable is categorical, branches will be created for each value taken by the feature. Otherwise, if the variable is continuous, the split will be done on a specific threshold value $t_V^{(k)}$, and two branches will be created for values greater or smaller than this value.

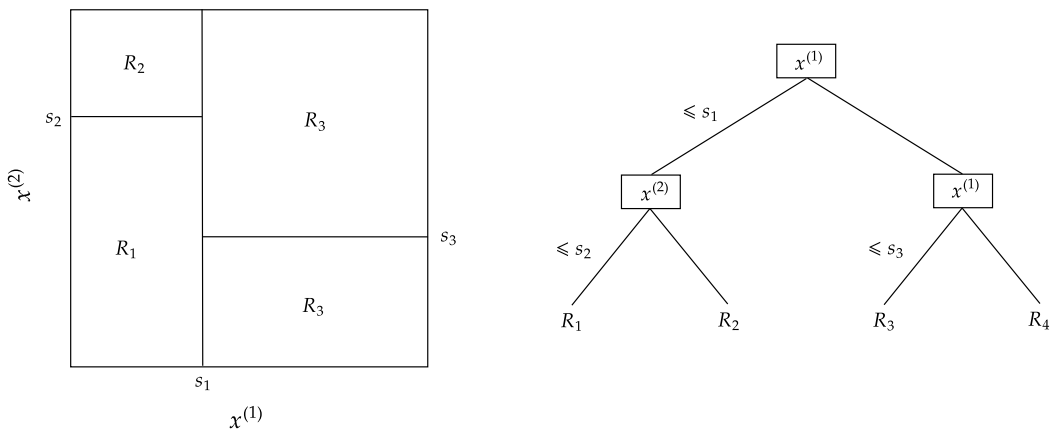


Figure 2.2: This figure illustrates a 2D representation of a decision tree structure. On the right-hand side, the decision-making process is depicted as a tree structure, where each node represents a splitting criterion. On the left-hand side, the tree is represented in a 2D square, showcasing the distinct regions corresponding to the decision boundaries of the leaf nodes.

CART Algorithm The best-known algorithm for decision trees is CART (classification and regression tree) proposed by Breiman et al. (1984). It is an iterative algorithm that constructs a tree-like structure by recursively partitioning of the training data $\mathbf{x}$ into smaller subsets based on a splitting criterion. This iterative process aims to create homogeneous subsets, maximizing the separation between distinct categories. Each leaf node in the resulting decision tree represents a region in the feature space that is associated with a specific class. Figure 2.2 illustrates two common representations of decision trees in the 2D case: the classic tree structure on the right and the 2D partitioning diagram on the left. In general, by supposing that we have a partition of the input space $\mathbb{R}^P$ into M regions $R_1, \ldots, R_M$ where $R_i \cap R_j = \emptyset$ for $i \neq j$, each associated with a class value $c_m \in \{1, \ldots, K\}$, the regression model is expressed as (Tibshirani et al., 2021) :

$$T(\mathbf{x}) = \sum_{m=1}^{M} c_m \mathbb{1}(\mathbf{x} \in R_m) \tag{2.9}$$

where $\mathbb{1}$ is the indicator function. Let D_m be the training samples that belong to the leaf node associated to region R_m and $|D_m|$ be the number of samples. In classification, c_m can be chosen as the most frequent class $c_m = \underset{c \in \{1,\ldots,K\}}{\operatorname{argmax}} \sum_{i, y_i \in D_m} \mathbb{1}(y_i = c)$. In regression, minimizing the sum of squares yields $c_m = \frac{1}{|D_m|} \sum_{i, y_i \in D_m} y_i$.

In the algorithm, the root node at the top of the tree encompasses the entire data set. At each subsequent node V, a splitting decision is made based on the most informative feature $X^{(k)}$ with a threshold $t_V^{(k)}$, resulting in the separation of data into two child nodes. Hence, each node V contains a subset of the training vectors, where their number is denoted as $|V|$. This process continues recursively using these two steps:

- the left subset V_l which contains all values $x^{(k)} \leq t_V^{(k)}$ where the proportion is $p_l = |V_l|/|V|$

- the right subset V_r which contains all values $x^{(k)} > t_V^{(k)}$ where the proportion is $p_g = |V_g|/|V|$

To effectively partition the data and identify optimal splitting points, decision trees employ impurity measures. They quantify the homogeneity within a subset allowing the algorithm to choose the split that leads to the most homogeneous subsets. For each variable and each split point, CART selects the one that minimizes the impurity, thereby promoting homogeneous subsets. The optimization problem is defined as the minimization of the expectation of the impurity measure:

$$\underset{k\in\{1,\dots,p\};t_V^{(k)}\in\mathbb{R}}{\operatorname{argmin}} \quad \mathbb{E}[I(V)] \tag{2.10}$$

where $\mathbb{E}[I(V)] = p_l I(V_l) + p_r I(V_r)$.

The algorithm works by recursively splitting for each node and for each subset V_l and V_r until a stopping criterion is met, such as reaching the maximum depth of the tree, a minimum number of samples, or a threshold on impurity measure.

Impurity measure Let us take a classification problem and consider a node V, represented by a subset of the data set $\mathcal{D}_V$ containing $|V|$ observations. Let us define the proportion of class $c \in \{1, \dots, K\}$ observed in the node V:

$$p_{V_k} = \frac{1}{|V|} \sum_{i,\mathbf{x}_i\in\mathcal{D}_V} \mathbf{1}(y_i = c) \tag{2.11}$$

The CART algorithm uses two main impurity measures:

- **Gini impurity** quantifies the probability of misclassifying a randomly selected instance from the subset based on the distribution of labels within the subset. A higher Gini Index indicates greater heterogeneity or impurity, suggesting that the subset can be further partitioned to enhance homogeneity. It is expressed as:

$$I(\mathcal{D}_V) = 1 - \sum_{k=1}^{K} p_{V_k}^2 \tag{2.12}$$

  This method is not tailored for continuous features or discrete variables with multiple values.

- **Information Gain** measures the reduction in entropy achieved by splitting the data based on a particular feature. A higher information gain indicates a more informative split, as it indicates that the split is able to effectively separate the data into more homogeneous subsets. The Information Gain is expressed as:

$$IG(\mathcal{D}_V) = H(\mathcal{D}_V) - \sum_{V'} p_{V'} H\left(\mathcal{D}_{V'}\right) \tag{2.13}$$

  where the entropy is defined as

$$H(\mathcal{D}_V) = -\sum_{k=1}^{K} p_{V_k} \log\left(p_{V k}\right) \tag{2.14}$$

and $\mathcal{D}_V$ representing the training data at node V. Other metrics could be used, including entropy measure and classification error. Note that for regression tasks, CART uses mean squared error as a metric.

CART analysis involves four main steps (Lewis, 2000):

1. Tree Building
2. Stopping Criterion
3. Pruning: Candidate tree generation by pruning and nodes removing to avoid overfitting
4. Optimal Tree Selection: Selection from pruned trees the one that best matches training data

Variants of CART There are different variants of the CART algorithm that have distinct characteristics, particularly in terms of data handling capabilities, stopping criteria, pruning strategies, impurity measures, and optimal tree selection. Notable examples of these variants include:

- **ID3** (Iterative Dichotomiser 3) developed by Quinlan (1986), is the pioneering decision tree algorithm designed specifically for categorical variables and classification tasks. It employs a multiway tree structure, where nodes can branch into more than two child nodes. ID3 utilizes information gain or entropy to identify the most informative split at each node, to build the tree.
- **C4.5** (Quinlan, 1993) extends the applicability of ID3 to continuous data by discretizing numerical attributes into a set of intervals. Additionally, C4.5 employs a more refined pruning strategy, evaluating the accuracy of the branches of the tree and eliminating them if they do not improve accuracy. C5.0 is an improvement of C4.5 in terms of algorithm efficiency and accuracy.

Decision trees offer the advantage of being interpretable with simple rules that are easy to understand and visualize. They can tackle both continuous and categorical data variables. However, the simplicity of these models can hinder their effectiveness when dealing with complex and non-linear relationships. Moreover, as the number of variables increases, understanding the logic behind a large decision tree becomes challenging. Sensitivity to small data changes is a major drawback of decision trees. These changes can lead to significant alterations in the tree's structure, impacting its understanding.

Hybrid methods

Beyond the basic structures, various approaches build upon decision rules and trees. Methods such as SIRUS (Bénard et al., 2021) aim to take advantage of the accuracy of the Random Forest algorithm (Breiman, 2001) and extract interpretable rules. Rule fit (Friedman & Popescu, 2008), for example, merges their interpretability with linear models, learning a rule combination from trees as features. Bayesian rule lists (Letham et al., 2015) combine rules with probabilistic frameworks, offering uncertainty quantification. Fuzzy logic (Chen et al., 2018; Zadeh, 1965), another rule framework presented in section 3.3.4, aiming to mimic human language and handle

uncertainty, extends both trees and rules. These advancements showcase the versatility and adaptability of these foundational methods.

Following the description of decision rule methods, we now describe Bayesian networks, a technique that provides a more flexible framework for representing variables and which deals with uncertainty through probabilistic reasoning.

2.4.2.3 Bayesian Networks

Probabilistic graphical models are powerful tools for representing and reasoning about complex relationships and uncertainty in data. They offer a different interpretation than tree structure, where the nodes represent the variables, and edges connecting these nodes denote conditional dependencies, capturing the influence variables have on each other. Bayesian Networks (BNs) developed by Pearl (1985, 1988), are probabilistic graphical models that represent a set of random variables and their probabilistic relationship using graph theory. Judea Pearl, who introduced BNs, referred to them as "belief networks" because he viewed the probabilities in the networks as degrees of belief. They are defined as follows:

Definitions

Definition 2.4.5 (Bayesian Networks)**.** *A Bayesian Network is defined by:*

- *a* ***Directed Acyclic Graph*** *(DAG) $\mathcal{G} = (V, E)$, where $V = \{X^{(1)}, \ldots, X^{(p)}\}$ is the set of nodes representing the random variables and E is the set of edges.*
- *a set of* ***local conditional probability distribution*** *where each node $X^{(i)}$ is associated with its parents in the graph through the edge by a set of conditional probabilities $P(X^{(i)}|Parents(X^{(i)}))$.*

Hence, the joint distribution of a Bayesian Network factorizes according to the graph structure:

$$P(X^{(1)}, X^{(2)}, \ldots, X^{(p)}) = \prod_{k=1}^{p} P(X^{(k)}|Parents(X^{(k)})) \tag{2.15}$$

Remark 2.4.6. *Acyclic means that there is no directed cycle in the graph, meaning that there is no path from a variable back to itself. The absence of cycles also makes it possible to identify causal relations in the graph more easily, as the direction of influence from one variable to another can be determined.*

D-separation

From the graphical perspective, **d-separation** (Pearl, 1988) is a criterion for deciding, from any DAG $\mathcal{G}$, whether a set of vertices X of variables is independent of another disjoint set Y, given a third disjoint set Z.

Definition 2.4.7 (d-separation (Pearl, 1988))**.** *If X, Y and Z are three disjoint subsets of nodes in a DAG $\mathcal{G}$, X and Y are d-separated given Z in $\mathcal{G}$ if and only if there is no undirected path between X and Y along which the following two conditions holds:*

1. *every node in the path with a converging arrow is either in Z or has a descendent in Z*
2. *every other node on the path is not in Z*

*The path between X and Y is said to be **blocked**.*

This concept is based on the idea of blocking or making inactive a certain path in a graph, also existing in graph theory (Bishop, 2006). Indeed, d-separation aims to identify the configuration of nodes called **collider** (A node X with two incoming edges $X \rightarrow Z \leftarrow Y$). When conditioning on a collider, the path is active, and when conditioning on a non-collider, the path becomes blocked.

Markov Condition

The Markov condition is a property that describes the relationship between a node and its parents in a Bayesian Network. A Bayesian network must satisfy the Markov condition, which is a fundamental principle that facilitates the representation and computation of the joint probability distribution in BN.

Definition 2.4.8 (Markov Condition)**.** *Consider a joint probability distribution P of the random variables in a set V. The Markov condition for a Bayesian network with respect to $\mathcal{G} = (V, E)$ is expressed as :*

$$X^{(i)} \perp\!\!\!\perp NonDescendants(X^{(i)}) | Parents(X^{(i)}) \tag{2.16}$$

for all $X^{(i)} \in V$ where $\perp\!\!\!\perp$ represent independence. In other words, given the parents of $X^{(i)}$, the variable $X^{(i)}$ becomes conditionally independent of all non-descendant variables.

If the Markov condition is satisfied, we say that G and P satisfy the Markov condition with each other. Equivalently, if a Bayesian Network satisfies the Markov condition, then every d-separation identified in the DAG implies a conditional independence in the probability distribution.

Faithfulness

An important property in Bayesian networks is faithfulness. This concept describes the relationship between the network's structure and the probability distribution it represents.

Definition 2.4.9 (Faithfulness (Neapolitan, 2004; Spirtes et al., 2001))**.** *Consider a joint probability distribution P of the random variables in a DAG $\mathcal{G} = (V, E)$. P and $\mathcal{G}$ are said to be **faithful** to one another if the two conditions hold:*

- *$(\mathcal{G}, P)$ satisfies the Markov condition ($\mathcal{G}$ entail only conditional dependencies in P)*
- *All conditional independences in P are entailed in $\mathcal{G}$, using Markov condition.*

In simpler terms, faithfulness ensures that the network's structure accurately reflects the conditional independence relationships present in the distribution, and d-separation is the tool for identifying these relationships in the DAG.

Markov Equivalent Class

Usually, there is not a unique way to represent the conditional independence relationship in the form of a graph. In fact, two Bayesian Networks may have different graphical structures but present the same set of conditional independence relationships. In this case, we say that they belong to the same **Markov Equivalent Class**.

Graph Construction

Learning a Bayesian network with a DAG $\mathcal{G}$ and parameters Θ from a data set $\mathcal{D}$ involves two mains steps:

- Parameter learning: estimation of the values of the parameters (i.e. conditional probabilities distributions) in the network from data.
- Structure learning: construction of the graphical structure

which are expressed as follows (Scutari et al., 2018):

$$\underbrace{P(\mathcal{G}, \Theta|\mathcal{D})}_{\text{learning}} = \underbrace{P(\mathcal{G}|\mathcal{D})}_{\text{structure learning}} \cdot \underbrace{P(\Theta|\mathcal{G}, \mathcal{D})}_{\text{parameter learning}} \tag{2.17}$$

The following provides a description of the methodologies used in the two primary phases of Bayesian networks learning.

Parameters learning in Bayesian networks involves estimating the parameters that characterize the conditional probability distributions of each node variable given the values of its parents.

There are two main approaches:

- Maximum likelihood estimation (MLE): The MLE estimates the parameters by maximizing the likelihood function, which is the probability of observing the data given the network structure and parameter values. Given a data set $\mathcal{D}$, the likelihood is expressed as :

  $$L(\Theta|\mathcal{D}) = P(\mathcal{D}|\Theta) \tag{2.18}$$

  where $L(\Theta|\mathcal{D})$ is the likelihood function. The maximization is done using numerical optimization techniques such as gradient descent.

- Bayesian Estimation: it uses Bayes theorem to infer the posterior distribution of the parameter Θ given the data $\mathcal{D}$ and the prior knowledge about the network. The posterior distribution is expressed as:

  $$P(\Theta|\mathcal{D}) \propto P(\mathcal{D}|\Theta)P(\Theta) \tag{2.19}$$

The parameter can then be sampled from the posterior distribution using, for example, Markov Chain Monte Carlo (MCMC) methods.

In the following, we give a non-exhaustive overview of the structure learning methods in the literature. For more information, we recommend the reader to consult a detailed review in Kitson et al. (2023); Scutari et al. (2018).

Structure learning Learning the structure of a Bayesian network (BN) is a computationally challenging task, often classified as an NP-hard problem (Chickering, 1996). This is primarily due to the exponential growth in the number of possible network structures as the number of variables increases. Experts can manually construct BNs, which can be laborious and time-consuming, especially when dealing with large data sets or data sets with errors. Moreover,

the availability of experts with the necessary expertise and time commitment can be a limiting factor.

There are three main approaches in the literature (Kitson et al., 2023) to learn Bayesian networks structure from data:

- **Constraint Based Methods**: These methods typically involve two main steps. The first step is about statistical testing to establish the conditional dependencies between the variables and thus obtain the skeleton of the graph, i.e. the undirected graph structure. The second step then constrains the types of existing relationships and orients the edges of the graph. These relationships are assessed using various statistical tests such as the χ^2 test and mutual information. The assumption of causal sufficiency is often made, implying that all relevant causal factors have been measured and included in the data set, ensuring that the identified relationships reflect true causal dependencies in the system. These methods usually return the set of DAG from the Markov equivalent class.

  Kitson et al. (2023) divide the methods into three types:

  - *Global discovery algorithms*: The goal is to learn the whole graph structure. Algorithms like PC algorithm (Spirtes & Glymour, 1991), which starts with a fully connected graph and removes edges, and the SGS algorithm (Spirtes et al., 2001), which on the opposite starts from scratch and constructs the graph. Several improvements and different versions of these algorithms were made.
  - *Local discovery algorithm*: These methods learn the skeleton related to each variable separately, which means that they consider the dependence on the neighbor node. They are then merged to create the overall graph. For example, the Grow-Shrink algorithm (Margaritis & Thrun, 1999) consists of two phases, one attempting to add edges (grow) and one attempting to remove edges (shrink) from the graph.
  - *Latent Variables Algorithm*: These methods do not assume causal sufficiency and deal with latent variables. The methods developed introduce new types of graphs that allow us to take into account these latent variables without making the problem intractable. The most popular method is the FCI algorithm developed by Spirtes et al. (1993, 2001).

- **Score-base methods**: These methods aim to find the best structure in a DAG space based on scoring. Score-based models aim to solve the following problem :

$$G* = \underset{G \in \mathcal{G}}{\operatorname{argmax}} \ \text{Score}(G, \Theta|\mathcal{D}) \tag{2.20}$$

  where $\mathcal{G}$ represents the set of all possible graph structures and $Ttheta$ the parameters of the graph. According to Kitson et al. (2023), the method consists of two elements:

  1. a *search strategy*: The search strategy used by score-based methods operates by defining the search space, which encompasses the set of possible network structures to be explored. This could involve considering all possible DAGs or focusing on a subset representing the Markov equivalence class of the true network structure. Once the search space is defined, a heuristic search algorithm is used to navigate through the possible structures, aiming to identify the one with the highest score. Various heuristic search algorithms exist, including greedy search, hill climbing, and

evolutionary algorithms. These algorithms can be broadly categorized into two types: approximate and exact algorithms. Approximation algorithms such as K2 (Cooper & Herskovits, 1992) or HC algorithm (Heckerman et al., 1995) aim to find a high-scoring structure within a reasonable amount of time, but they may not guarantee the optimal score. Exact algorithms such as CPBayes (Van Beek & Hoffmann, 2015) or GOBNILP-DEV (Liao et al., 2019), on the other hand, guarantee the identification of the structure with the highest score, but they may be computationally more expensive and less scalable.

2. a *score function*: There are two types of functions in the literature.
   - *Bayesian scores* provide a measure of the relative likelihood of a given network structure, considering both the data and the prior beliefs about the network's structure and the relationships between variables in scores such as K2 (Cooper & Herskovits, 1992) and BDeu (Buntine, 1991; Heckerman et al., 1995).
   - *Information-theoric score* provides a measure that assesses how well the structure fits the data while avoiding overfitting with a penalty term preventing complex models. Common metrics include metrics such as Bayesian Information Criterion (BIC) (Schwarz, 1978), Minimum Description Length (MDL) (Suzuki, 1999), Akaike Information Criterion (AIC) (Akaike, 1974) and quotient Normalised Maximum Likelihood (dNML) (Silander et al., 2018).

- **Hybrid methods**: These methods combine aspects of constraint-based and score-based approaches to achieve the best of both worlds. They aim to harness the strengths of constraint-based methods, such as the ability to deal with high-dimensional data sets, by restricting the search space and guiding the learning process while also benefiting from the goodness-of-fit maximization capabilities of score-based methods. This combination allows hybrid methods to overcome the limitations of individual approaches. Several algorithms were developed in this sense, such as the Sparse Candidate algorithm (Friedman et al., 2013) and Max-Min Hill Climbing algorithm (Tsamardinos et al., 2006).

As we conclude our discussion on Bayesian networks, the next part introduces symbolic regression, another approach that seeks to uncover the underlying relationships between variables in a data set.

2.4.2.4 Symbolic Regression

Symbolic regression (SR) (Koza, 1994; Schmidt & Lipson, 2009) is a machine learning approach that aims to uncover the underlying mechanism by discovering symbolic mathematical expressions that best describe relationships within data. They are particularly useful for extracting an interpretable predictive model and uncover relationships for scientific discovery and modeling.

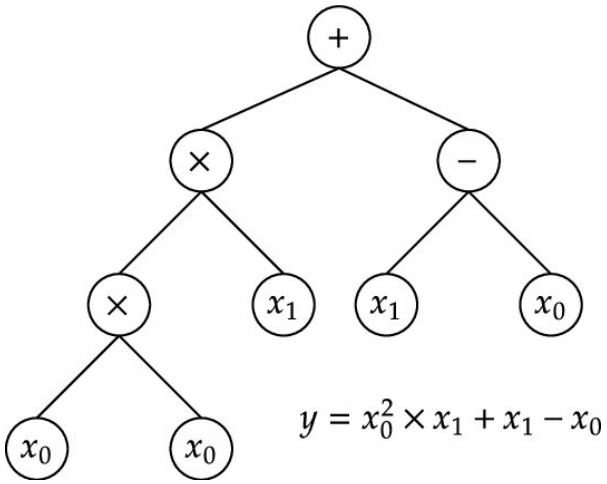


Figure 2.3: Figure from Jeschke et al. (2023)

The objective in SR is to find a function $\hat{f}$ over a class of function $\mathcal{F}$ that minimizes a loss function $\mathcal{L}$. The key distinction between SR and classical regression methods lies in the form of the function class $\mathcal{F}$ (Makke & Chawla, 2024). In fact, SR seeks to discover functions by combining a predefined set of discrete operators and functions, such as arithmetic operations ($+$, $-$, $\times$, $\div$) and mathematical functions (e.g., log, sin, cos). The set of all possible functions that can be built from all the combinations defines the function space $\mathcal{F}$.

The minimization problem is expressed as follows:

$$\hat{f} = \underset{f \in \mathcal{F}}{\operatorname{argmin}} \sum_{i=1}^{N} \mathcal{L}(f(\mathbf{x}_i), y_i) + \lambda \mathcal{C}(f) \tag{2.21}$$

where the loss function $\mathcal{L}$ can take various forms, such as the mean squared error.

While various symbolic regression methodologies exist, we focus on two main approaches (Camps-Valls et al., 2023; Makke & Chawla, 2024).

- **Discrete Search Methods**: Leveraging evolutionary principles, genetic programming iteratively evolves a population of candidate equations to discover the symbolic expression best fitting the data often represented as an expression tree. Through "breeding", high-performing individuals and introducing combinations and random variations (e.g. crossover, mutation), the algorithm efficiently explores the search space. Schmidt & Lipson (2009) developed this approach to discover physical laws from data, demonstrating its potential for scientific discovery.
- **Sparse Linear regression**: Traditional regression methods can be adapted to build symbolic regression approaches. One prominent example is Sparse Identification of Nonlinear Dynamics (SINDy), developed by Brunton et al. (2016). This approach expands the search space by transforming input variables using a library of base functions and interaction terms. By applying sparse regression, SINDy effectively selects the most relevant functions that accurately capture the system's dynamics.

Beyond these approaches, diverse approaches like AI Feynman (Udrescu & Tegmark, 2020) (inspired by physical principles like symmetry and separability) and reinforcement learning, along with neural networks, have emerged to tackle symbolic regression. For further exploration of these methods, we recommend the following resources: (Camps-Valls et al., 2023; La Cava et al., 2021; Makke & Chawla, 2024).

The true strength of symbolic regression lies in its ability to extract interpretable relationships, making it a valuable tool for scientific discovery and system identification. By providing equations that are readily understandable, it grants deeper insights into the underlying processes governing the data.

2.4.2.5 Local Interpretable methods

Naive Bayes is a popular classification algorithm based on Bayes' theorem. The "naive" assumption implies that the variables are conditionally independent given the class label y. In the discrete case, where the features take on a finite set of values, this is expressed as :

$$P(\mathbf{X}|Y) = P\left(X^{(1)}, X^{(2)}, \ldots, X^{(p)} \mid Y\right) = \prod_{k=1}^{p} P\left(X^{(k)} \mid Y\right) \tag{2.22}$$

Naive Bayes computes the posterior probability of the class y given an instance $\mathbf{x}$, $P(Y = y|\mathbf{X} = \mathbf{x})$, using Bayes' theorem. In the discrete case, where the features take on a finite set of values, the posterior probability is formulated as follows:

$$P(Y = y|X = \mathbf{x}) = \frac{P(Y = y) \cdot P(X = \mathbf{x} \mid Y = y)}{P(X = \mathbf{x})} \overset{\text{ind}}{=} \frac{P(Y = y) \prod_{k=1}^{p} P(X^{(k)} = x^{(k)} \mid Y = y)}{P(X = \mathbf{x})} \tag{2.23}$$

This equation highlights the interpretability of the model: the contribution of each feature $x^{(k)}$ to the class prediction is directly visible through its individual conditional probability term $P(X^{(k)} = x^{(k)}|Y = y)$. The class label is then predicted by choosing the class label from the set of classes $\mathcal{Y}$ that is the most likely given the data:

$$\hat{y} = \underset{c \in \mathcal{Y}}{\operatorname{argmax}} \; P(Y = c) \prod_{k=1}^{p} P\left(X^{(k)} = x^{(k)} \mid Y = c\right) \tag{2.24}$$

Naive Bayes offers inherent interpretability due to its simplicity and reliance on straightforward probability calculations. The model's output can be easily explained by presenting the conditional and prior probabilities involved. Analyzing these individual terms at the feature level provides insights into how specific features contribute to predictions. Moreover, it operates efficiently in high-dimensional scenarios.

However, the underlying assumption of feature independence is often unrealistic in real-world scenarios, leading to inaccurate probability estimations (Rish, 2001). Moreover, its expressiveness is limited, as it can only represent dependence through conditional probabilities, neglecting more complex relationships between features.

k-Nearest Neighbors (kNN) On the local level, the k-nearest neighbors algorithm offers a non-parametric approach for both classification and regression tasks. Notably, kNN is characterized

as a "lazy learner" due to its lack of training or fitting process. Instead, it focuses solely on storing the provided training data. When presented with a new instance $\mathbf{x}$, the kNN algorithm proceeds as follows:

1. Compute distance: compute the distance between $\mathbf{x}$ and every other sample in the training data set. Euclidean distance is a common choice for this metric.
2. k Nearest neighbors: select the k data points that have the smallest distance to $\mathbf{x}$, denoted as a set $S_k(\mathbf{x})$.
3. Prediction:
   - Classification: predict the majority class among the labels of the k nearest neighbors $\hat{y} = \underset{c \in \mathcal{Y}}{\operatorname{argmax}} \sum_{\mathbf{x}_j \in S_k(\mathbf{x})} I(y_j = c)$, where $I(\cdot)$ is the indicator function.
   - Regression: predict the average of the target value of the k nearest neighbors $\hat{y} = \frac{1}{k} \sum_{\mathbf{x}_j \in S_k(\mathbf{x})} y_j$

KNN poses two primary challenges for accurate predictions: selecting the optimal distance metric and determining the best value for the number of neighbors k. Regarding the former, various metrics exist, including Euclidean, Mahalanobis, and Minkowski distances, with their variants. Crucially, different applications necessitate different distance measurements for optimal performance. Secondly, finding the ideal k often requires cross-validation techniques (Zhang et al., 2017).

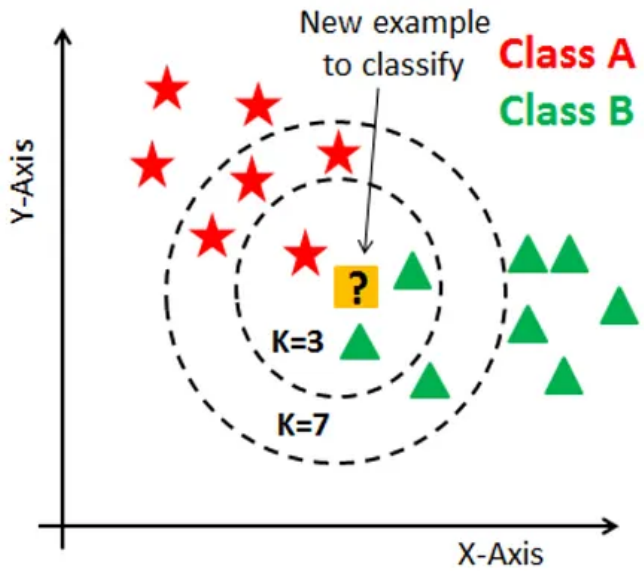


Figure 2.4: Figure from kdnuggets illustrating the classification of a new sample by the kNN algorithm

kNN is interpretable due to its reliance on distances to closest neighbors, promoting easy understanding. It offers simplicity and adaptability as its implementation is straightforward, requiring no training phase or parameter tuning. However, kNN suffers from the curse of

dimensionality. While visualizations like in Figure 2.4 provide insights in specific cases, understanding predictions based on averaged neighbors or majority class can become challenging, especially in high dimensions (Tibshirani et al., 2021). In such scenarios, the individual feature impact is lost with the collective neighbor's influence.

2.4.2.6 Discussion on interpretable models

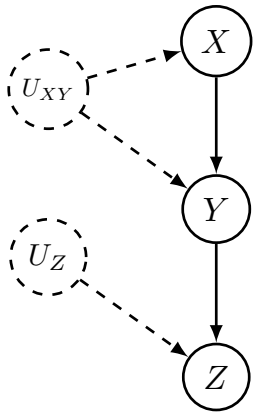
The importance of interpretability in machine learning models depends on the analysis goals (Burkart & Huber, 2021). Some models are inherently interpretable due to their simple structure. Hence, these models, such as decision trees only focus on achieving the best possible accuracy and performance. On the other hand, some models are interpretable by design, meaning that they are trained with specific constraints to add interpretability, such as penalized regression.

Further, some models offer both local and global interpretability. Regression models, for instance, reveal individual feature contributions (local) and overall trends (global). Conversely, local interpretability might be the only option for models like Naive Bayes or kNN, where understanding individual predictions is easier than grasping the entire model's behavior.

Choosing the right interpretability approach depends on your needs. If understanding each prediction is crucial, local interpreters like Naive Bayes or Bayesian networks might suffice. But for broader insights into model behavior, globally interpretable models like regression could be the better choice. It is important to remember that there's no one-size-fits-all solution – the best model depends on the specific objective.

2.4.3 Causality

Causality can be considered the ultimate goal in the field of XAI (Bhatt et al., 2020), as it aims to discover the true cause-effect relationships between the variables in a system and the mechanisms that link them. It allows us to fully understand the underlying relation that drives the observed phenomena, hence making informed and justified explanations and improving accuracy and reliance on ML models. Causality typically involves representing a causal graph which is either a **Causal Bayesian Network** (Pearl, 2000) (see section 3.3.6.1) representing the causal link or a **Structural Causal Model** (SCM) (Pearl, 2009b; Peters et al., 2017) (see section 3.3.6.2) representing the causal model of the data-generating process, as illustrated in figure 2.5. They differ from Bayesian Networks (seen in part 2.4.2.3) as arrows in BN indicate dependence, which can be due to causation but also correlation. While CBN represents only causal links, SCM represents the causal structure of the data-generating process, modeling the links between variables and accounts for noise introduced by unmodeled variables through functions known as causal mechanisms. When analyzing causal relationships, Directed Acyclic Graphs (DAGs) serve as the primary tool, visualizing the causal links and directions between variables.



$$M = (\mathbf{V}, \mathbf{U}, \mathbf{F}, P)$$
$$\mathbf{V} = \{X, Y, Z\}$$
$$\mathbf{U} = \{U_{XY}, U_Z\}$$
$$\mathbf{F} = \begin{cases} f_X : X := 2U_{XY}, \\ f_Y : Y := X + U_{XY}, \\ f_Z : Z := 3Y + U_Z \end{cases}$$
$$P = \begin{cases} U_{XY} \sim \mathcal{N}(0,1), \\ U_Z \sim \mathcal{N}(0,1) \end{cases}$$

Figure 2.5: Figure from Zanga & Stella (2023) representing the causal graph on the left associated with a structural causal model (SCM) on the right where V and U are respectively the set of endogenous and exogenous variables, F is a set of functions and P is a joint probability distribution over the exogenous variables.

In the following, we describe the main concepts of causality alongside their main methods.

Causal Discovery The study of causality requires, first and foremost, the identification and modeling of cause-effect relationships between variables in a data set. In this regard, Causal Discovery aims to provide tools and algorithms in order to learn and differentiate between correlation and causal links, find patterns, and discover the latent influence of a variable that is not present in the data set. The discovery methods can be divided into different categories (Camps-Valls et al., 2023; Glymour et al., 2019; Peters et al., 2017):

- **Independence-base** (or Constraint-based) methods use marginal and conditional independencies between variables to infer the underlying causal structure. They do not assume any causal mechanism within the SCM, focusing only on the patterns of independence revealed by the data. Algorithms like PC (Spirtes et al., 2001) or FCI (Spirtes & Glymour, 1991) can identify causal structures from data under specific assumptions about the data. However, as seen in part 2.4.2.3, BN does not provide exact causal information as it outputs the Markov equivalent class, i.e., a set of structures satisfying the same conditional independence.

- **Functional-based** methods explore causal relationships by assuming a specific form for the underlying process in the SCM. These methods, such as LiNGAM (Shimizu et al., 2006), represent causal relationships through a parametric form where each variable is modeled as a function of its direct cause and where noise follows a specific distribution. These approaches allow taking advantage of the asymmetric nature of these functions to uncover the true causal direction, unlike dependence-based methods, by identifying the causal structure in the same equivalent class (Glymour et al., 2019).

- **Score-based** methods assign a scoring function to evaluate a causal structure based on how well it fits the data. They then employ search algorithms, like the popular Greedy

Equivalent Search (GES) (Chickering, 1996), to iteratively explore different structures and identify the one with the highest score, best fitting the data.

Causal Inference aims to estimate the causal effects of one variable on another, either by estimating the impact of interventions or by analyzing observational data under causal assumptions. The interventions are changes that are made in order to observe the effects that should be done in an experimental setting. Usually, causal inference aims to estimate accurately even in the presence of confounding factors and sources of bias in the data set. The estimation can be carried out using experimental design such as Randomized Control Design (RCT). When it is not possible, two main frameworks can be used:

- The **Potential Outcome Framework** developed by Neyman et al. (1990); Rubin (1974) formalize causal relationships by considering potential outcomes of variables under various treatment conditions. To accurately estimate causal effects in observational studies, methods like propensity score matching are used to create comparable groups with similar distributions of covariates between treated and control groups.
- **Do-Calculus**, developed by Pearl (2009b), is a framework for reasoning about causal relationships in observation data. It allows manipulating a causal graph, simulating interventions, and answering counterfactual questions, such as "What would have happened if?" to understand the true causal effects of one variable on another. This framework enables the extraction of true causal relationships when controlled experiments cannot be done.

For deeper insights into these concepts, which fall outside the scope of our study, we direct readers to the following resources (Acharki et al., 2023; Imbens & Rubin, 2015; Pearl, 2009a; Yao et al., 2021).

2.4.4 Conclusion on the state of the art

Causality can improve both explainability and interpretability. Understanding causal relationships in the data allows for a better understanding of the model's behavior and, hence, interpretation of its decision-making process in light of the true underlying mechanism. Similarly, causality can improve explainability by bringing a causal and logical explanation from the input to the model's decision, which is more intuitive and convincing. Conversely, interpretability and explainability can play a role in shaping the model that aims to extract causal relationships in a data set.

	Interpretability	Explainability	Causality
Definition	Degree of model understandability	Ability to provide insights into decisions	Identifying cause-effect relationships
Goal	Transparency	Decision clarity	Reveal true variable relationships
Challenges	Accuracy vs. simplicity	Complex model explanation	Confounding variables, data limitations

Table 2.4: Comparison of XAI approaches: Interpretability, Explainability, and Causality

XAI is an important field that aims to be present in every field of Machine Learning in order to ensure that the models are interpretable, explainable, or causal, allowing them to act in a reasoned and justified manner. It answers multiple questions such as:

- Why this prediction?
- What factors influenced the outcome?
- How does the model work internally?
- How can I improve the outcome?
- What would happen if I changed this input?
- What else could happen?

Once the XAI models have answered these questions, it becomes necessary to assess and evaluate their effectiveness and whether these answers correspond to reality and expectations. In the following, we focus only on interpretability and causality as they will be the main focus throughout this thesis.

2.5 Evaluation

The literature highlights several reasons for evaluating XAI methods (Markus et al., 2021; Zhou et al., 2021). One key benefit is that it allows for the comparison of different XAI models, enabling researchers to choose the best one for a specific task. This evaluation process involves assessing how well each model satisfies the properties of interpretability, explainability, or causality. Additionally, evaluation helps to determine if an XAI model achieves its intended objective in a real-world setting.

Indeed, the establishment of an interpretable or causal model inevitably raises fundamental questions: Is the explanation provided satisfactory? Is it comprehensive enough? How faithfully does the explanation represent the underlying system and reality? And how do we assess the validity of the proposed explanation? Given the interdisciplinary nature of XAI methods, a large body of research has been devoted to addressing these questions.

One of the foundations for answering these questions lies in the understanding of human psychology and what constitutes a good explanation for a person. In this regard, the fields of philosophy, psychology, and cognitive science sought to explain how humans generate, evaluate, accept, and rely on explanation (Lopes et al., 2022; Miller, 2018). These studies are necessary to establish explanations that are adapted and useful to the user. For instance, Miller (2018) analyzed the literature in science and identified that probability, simplicity, generality, and coherence with prior belief are the most important criteria that people use to evaluate explanations.

In this section, we first present the different levels of evaluation, then the real limitations that are faced when evaluating interpretable and causal models, and, finally, we explore the different evaluation processes and objective criteria used in the literature.

2.5.1 Levels of evaluation

The objective of the evaluation is mainly determined by the stakeholders involved and the domain of application. From the XAI literature, the answer to give to previous questions depends on many factors. Researchers commonly use Doshi-Velez & Kim (2017) taxonomy which categorizes human subjects categorization rather than models:

The Doshi-Velez & Kim (2017) taxonomy, commonly used in the literature, provides different levels for evaluating XAI models:

- **Application-grounded evaluation**: involves human experts evaluating the explanations in the context of a specific application. The baseline is typically a human's explanation for the same task. This method is the most specific but also the most expensive and time-consuming.
- **Human-grounded evaluation**: involves human experiments that are usually layman person. The focus is on the quality of the explanation, regardless of the model's accuracy in the target domain. This method is less specific than the first, but still relatively expensive.
- **Functionally-grounded evaluation**: does not involve human experiments and evaluates the model's inherent characteristics like transparency or simplicity. This method is the least specific and cheapest, but it may not capture how humans actually understand the model.

Note that the specificity and cost of each approach increase as we use human and particularly domain experts. Similarly, Lopes et al. (2022); Zhou et al. (2021) regroup the first and second categories into human-centered evaluation and call the third category computer-centered evaluation.

The first aspect of evaluating explanations lies in a human-centered objective. The selection of a group of humans for experimental tasks allows for the evaluation of the explanation provided by the XAI model. In fact, qualitative metrics are evaluated such as user's trust, explanation usefulness or satisfaction, understandability or performance that is provided by the explanation through interviews or questionnaires (Lopes et al., 2022; Zhou et al., 2021). In addition, quantitative metrics are evaluated, such as the impact of a number of features and transparency of the model on the user's acceptance of explanation (Poursabzi-Sangdeh et al., 2021) or response time of the user's decision, where fast and accurate decision indicates a good understanding (Schmidt & Biessmann, 2019). The advantage of these approaches is that they provide direct and strong evidence for success (Doshi-Velez & Kim, 2017) but suffer from the subjective nature of the experiment. In addition, the process is time-consuming and expensive. On the contrary, functionally-grounded (computer-centered) evaluation does not require human experiments and can provide objective quantitative metrics.

In the following, we focus on the Functionally-grounded evaluation.

2.5.2 Evaluating Interpretability and Causality

While interpretability and causality share common goals, they also address specific questions. Table 2.5 illustrates the difference between these two areas by giving examples of questions they tackle.

	Interpretability	**Causality**
Transparency	Can the decision-making process be understood?	Does the model represent the underlying system?
Feature Interactions	How do features influence each other?	Can causal relationships between features be identified?
Instance-Level Explanation	Why does the model predict a specific outcome?	What is the causal impact of specific features on the outcome?
Accuracy	How does an interpretable model's accuracy compare to complex models?	How accurate is the causal model in capturing real-world relationships?

Table 2.5: Example of questions for interpretability and causality research in XAI

Evaluating interpretable and causal models necessitates a comprehensive approach that considers both the specific questions being asked and the context of the application. In the next sections, we describe general quantitative evaluation existing in the literature.

2.5.3 Quantitative evaluation in the literature

Quantitative evaluation of XAI models is a challenge in the literature as there is a lack of standardized approaches with a large range of models and various metrics (Lopes et al., 2022). Nevertheless, several authors propose a taxonomy based on the properties of the XAI model in order to do the evaluation, they are composed of two main categories (Lopes et al., 2022; Markus et al., 2021; Zhou et al., 2021). First, the explanation should be understandable to humans and be evaluated based on the following characteristics :

- clarity: the explanation should be unambiguous and similar for several instances. For example, Lakkaraju et al. (2017) quantifies it with a metric that calculates the number of feature instances that can have multiple targets in a rule.
- broadness: the explanation should be generally applicable. Nguyen & Martínez (2020) uses the feature mutual information to measure this dimension.
- simplicity/parsimony: explanation should be presented in simple and compact form. A large body of work evaluates this aspect using metrics such as model size (Lakkaraju et al., 2017), tree depth (Ribeiro et al., 2016), and runtime operating count (Slack et al., 2019).

Secondly, the explanation should have fidelity, meaning that it should accurately describe the model behavior in the entire feature space with two main properties:

- Completeness: the explanation should describe the entire dynamic of the model, i.e., how many input features that are used in the decision-making process are captured in the explanation.
- Soundness: the explanation should be correct and truthful to the model's task.

Note that by essence, interpretable models have the fidelity property with both completeness and soundness.

Nauta et al. (2023) propose a deeper taxonomy with an extensive review of the methods that fall into each of the categories. Other metrics are evaluated, such as the **coherence** metric, which includes the alignment with domain knowledge when the ground truth is available, and the continuity metric that assesses the stability of the model to slight perturbation.

We invite the reader to consult the following surveys for an in-depth study of the metrics used in the literature (Lopes et al., 2022; Nauta et al., 2023; Zhou et al., 2021).

2.5.4 Evaluation in practice

Since no single evaluation criteria or metric applies to all XAI methods, the performance of interpretable and causal models is usually evaluated based on the specific tasks they were designed for, such as forecasting accuracy, classification precision, and prediction error. While the focus is on evaluating their ability to capture the underlying mechanisms and relationships within the data, interpretability, and causality, introduce valuable constraints, promoting simpler models that foster understanding. By carefully validating interpretable and causal models with domain-specific metrics and tasks, researchers aim to ensure their models produce meaningful insights that align with established knowledge within the field, ultimately enhancing their usefulness in various applications.

2.6 Challenges & conclusion

Explainable AI has become an urgent need for integrating machine learning models into real-world applications. Despite its undeniable ambition and potential, XAI currently lacks agreement on its definition. This chapter addressed this ongoing debate, ultimately establishing the definition that will be used throughout the remainder of this thesis. Furthermore, we delved into various XAI models, particularly interpretable and causal ones, using taxonomies to understand their multifaceted nature. While promising approaches exist for various tasks, there is no "one-size-fits-all" solution due, for instance, to diverse objectives and data modalities. Consequently, no standardized procedure exists for measuring and comparing XAI models. Our exploration of evaluation methods showed the importance of assessing XAI models from different angles, beyond just accuracy metrics, which currently dominate the literature's evaluations.

We emphasized the importance of understanding stakeholder needs and the intended purpose of explanations when developing XAI models. In this objective, Bhatt et al. (2020) identified and offered valuable recommendations to select the model: 1) identify relevant stakeholders, 2) understand their needs, and 3) clarify the purpose of the explanations.

Recognizing the need to advance the XAI field, this thesis tackles key challenges identified in the literature, including uncovering causal relationships within complex data, enhancing

the overall performance of XAI models, and expanding XAI's approaches with interpretable models (Bhatt et al., 2020). Following Rudin (2019) objectives, the core focus of the following chapters lies in proposing interpretable models that are reliable and accurate, ultimately aiming to extract causal insights directly from the proposed models.

CHAPTER 3

Analysing Time Series: Uncovering Patterns and Influencing factors

Contents

3.1 Introduction to Root Cause Analysis

In order to improve their productivity and remain competitive in the market, companies in various sectors are developing increasingly sophisticated and expensive tools. In general, safety, reliability, and performance constraints are high, and the appearance of abnormal situations must be dealt with quickly, especially in critical sectors such as automotive, industry, or aeronautics, to avoid major damage or material and financial losses.

Root cause analysis (RCA), also known as fault diagnosis or fault identification, uses various terminology in the literature to precisely identify the underlying reasons behind system malfunctions. In the following sections, we present the terminology defined by Solé et al. (2017):

- **Events**: Exceptional conditions, which are occurrences that deviate from normal operating conditions of the system.
- **Fault/Problem/Root cause**: Events that can trigger subsequent events but are not themselves caused by other events.
- **Errors**: Events caused by the fault.
- **Failures**: Errors that can be observed externally from the system, often detected using anomaly detectors.

Upon identifying system failures, root cause analysis aims to determine the root causes of these anomalies from data collected from monitoring. Various methodologies have been developed for this purpose (Solé et al., 2017). The first, which is the most costly in terms of time and human resources, involves calling on a group of experts in the field to develop a model to determine the causes based on historical data. While these models can be highly accurate thanks to their expertise and understanding of the phenomena involved, in a highly complex system where physics models cannot explain all the variations, this approach becomes limited. Numerous traditional methods are used to find them, such as Pareto analysis, Ishikawa & Ishikawa (1982)'s cause-and-effect diagram or the Five Whys (Ohno, 2019), but they are limited by their strong dependencies on human skills, making the process time consuming, and by their subjectivity and bias. In addition, they cannot determine complex causes, where, for example, numerous variables influence the problem in non-linear ways.

The second type of approach, which contains most RCA methods in the industrial literature (Sayed & Lohse, 2013; Solé et al., 2017), consists of taking advantage of the presence of manually specified models that describe sub-parts of the system. They are usually developed using the first approach by the experts. With the help of domain knowledge, an RCA model is built by assembling the models of the sub-parts.

Finally, in the absence of domain knowledge, particularly in new or complex systems, an RCA model is constructed using available data. This approach is particularly useful as it does not require human intervention. The algorithms in this data-driven approach aim to extract patterns and relationships from the data, allowing for the discovery of the root cause.

With the increasing storage capacity and the growing amount of available data, artificial intelligence has become a powerful tool for RCA. This integration empowers RCA models to leverage massive data sets and use automated and sophisticated techniques to extract valuable insights and improve root cause identification.

In the following, we only focus on the last category, i.e. machine learning methods for root cause analysis when there is no prior domain knowledge about the system.

3.2 Challenges and problem statement

3.2.1 Objective and challenges

In the field of root cause analysis, the depth and scope of the analysis are contingent upon the types of questions posed. These questions shape the collection, selection, and analysis of data. The desired level of detail, i.e., granularity, significantly impacts the nature of the data needed. In the following, we describe the different characterizations of the root cause that are treated in the literature. Oliveira et al. (2022) research identifies three key questions that root cause analysis (RCA) aims to address:

1. **Localization of the root cause: Where is the root cause?** The objective is to identify the specific location and timestamp of the occurrence of the root cause by providing answers such as "The problem occurred in component k at time T". However, this type of analysis does not uncover the underlying root cause variable.

2. **Local identification of root cause: What caused the failure?** The aim is to identify a tangible effect of the true underlying cause and to provide a comprehensive explanation that elucidates how the root cause triggered the problem, such as "The issue occurred because the variable $X^{(k)}$ surpassed the threshold η.".

3. **Global identification of the root cause: Why did the root cause happen?** The aim is to go beyond identifying the root cause by delving into the underlying reasons for its occurrence and, if multiple causes exist, determine the relationships between them. This involves uncovering the factors that led to the root cause's deviation from its normal state, ultimately causing the failure, such as "The root cause $X^{(k)}$ exceeded the threshold η because the operator has not performed the update.".

Each question targets a specific aspect of the problem and offers varying levels of explanation. In addition, Oliveira et al. (2022) introduces three distinct types of data: location-time data, typically presented in tabular form (Component, Time), physical variables data, primarily comprising numerical values but potentially incorporating categorical variables, and log action data, consisting of recorded user actions.

Root cause analysis can be a challenging task due to several factors (Papageorgiou et al., 2022; Solé et al., 2017):

- **Scale and complexity of data**: RCA often involves analyzing vast amounts of data, which may include a wide range of features and a large number of observations. It can, therefore, be difficult to identify patterns and relationships that might reveal root causes.

- **Data Quality**: Identifying the root cause necessitates a comprehensive and reliable data set. However, several factors can impede the effectiveness of RCA, including data scarcity, missing values, excessive noise, and inconsistent sampling rates.

- **Rare fault occurrences**: Faults may be rare and sporadic in the training data, which can make it difficult to develop effective RCA models. This may require the use of techniques such as anomaly detection or outlier analysis to first label failures and then identify potential root causes.
- **Real-time requirements**: In some contexts, RCA needs to be carried out in real-time, imposing additional constraints on the analysis process. This may require the use of models capable of processing large data sets quickly and efficiently.
- **Limited universality of workflow**: There is no single, universally applicable workflow for RCA. The specific approach will vary depending on the type of incident, the available data, the business context, and the required domain and system knowledge.

In addition, Papageorgiou et al. (2022) highlights the challenge of developing XAI models that enable users to understand the decision-making process and the underlying reasoning behind the decision. This would reinforce trust and confidence in the model, as users would be able to understand the rationale behind the conclusions drawn.

In what follows, we focus exclusively on methods for dealing with data sets encompassing numerical data and, potentially, categorical variables. This study specifically addresses the scenario where we have labeled failures in a data set and aim to identify their underlying root causes.

3.2.2 Problem definition

In order to comprehensively explore the literature on root cause analysis and understand the diverse approaches, we first formulate the problem. This structured framework serves as a foundation for analyzing the literature and identifying the fundamental concepts, methodologies, and techniques used.

To ensure consistency throughout the thesis, we adopt the same notation introduced in Chapter 2. This allows each chapter to be read independently while maintaining a unified framework.

Consider N observations of p variables from monitoring systems, such as sensor data. Let $\mathbf{X} = (X^{(1)}, \ldots, X^{(p)})$ be a random vector of dimension $p \in \mathbb{N}$ where for each $k \in \{1, \ldots, p\}$, $X^{(k)}$ takes its values in $\mathbb{R}$ and let Y denote the random variable representing the failure occurrence. Y takes its values in a domain $\mathcal{Y} \subseteq \mathbb{R}$, which can be a finite or a continuous set for classification and regression problems respectively. The training data-set is represented as $\mathcal{D}_N = \{(\mathbf{x}_i, y_i) | i = 1, \ldots, N\}$, where each $(\mathbf{x}_i, y_i)$ pair is drawn from the joint distribution of $\mathbf{X}$ and Y and where $\mathbf{x}_i = (x_i^{(1)}, \ldots, x_i^{(p)})$. Additionally, let $\mathbb{X} = (\mathbf{x}_1^\intercal, \ldots, \mathbf{x}_N^\intercal) \in \mathbb{R}^{N \times p}$ be the design matrix and $\mathbf{y} = (y_1, \ldots, y_N) \in \mathcal{Y}^N$ be the target vector. In the following, for simplicity, we refer to $(\mathbf{x}, y)$ for a single instance where $\mathbf{x} = (x^{(1)}, \ldots, x^{(p)})$.

Consider that the following equation defines the real-world phenomenon that we analyze:

$$Y = f(\mathbf{X}, \epsilon)$$

where $f : \mathbb{R}^p \to \mathcal{Y}$ represents the underlying relationship between the input features and the target variables related to the failure, ϵ is the error term accounting for unobserved factors, measurements errors in the system or other sources of variability. In the literature, Y can be a

continuous measure associated with the problem, taking values in $\mathbb{R}$ or a binary variable taking value in $\{0, 1\}$, such as an alarm signaling the problem's occurrence.

The original data set often takes the form of time series data, which is preprocessed and transformed to incorporate temporal dependencies by introducing lagged variables. We will delve into various objectives commonly encountered in the literature to gain a deeper understanding of the mathematical formulation.

3.2.2.1 Root Cause Identification

In this analysis, the aim is to identify the root cause of a failure from numerical and categorical data obtained during data collection. In this setting, the target variable Y represents the occurrence of the failure. The goal of this root cause analysis is to learn a function $\hat{f} : \mathbb{R}^p \to \mathcal{Y}$ which is interpretable and associates the input observations with the identified failure. The resulting explanations provide insights into the model's understanding of the data and highlight the underlying relationship between the input features and the root cause.

3.2.2.2 Learning Process

The learning process can be broadly categorized into two main categories (Papageorgiou et al., 2022; Solé et al., 2017):

- **Deterministic**: it involves minimizing a loss function $\mathcal{L}$, such as the mean squared error, by adjusting the model parameters. The goal is to find a function $\hat{f}$ that minimizes the difference between the predicted failure occurrence $\hat{f}(\mathbf{x}_i)$ and the actual failure y_i for all observation i in the data-set.
- **Probabilistic**: the task typically involves estimating the conditional probability $P(Y = 1|\mathbf{X} = \mathbf{x})$, which represents the probability of observing failure $Y = 1$ given the observed feature vector $\mathbf{x}$. This can be achieved using methods like maximum likelihood estimation (MLE), which involves finding the parameter values that maximize the likelihood of the observed data.

In the following, we focus on the root cause identification problem using interpretable methods and examine the literature in this field. This analysis will provide us with a clear understanding of the current state of the art, allowing us to evaluate the strengths and limitations of existing methodologies.

3.3 State of the art

This section delves into a range of interpretable methods frequently used in root cause analysis (RCA). The methods we explore include regression analysis 3.3.1, association rules 3.3.2, decision trees 3.3.3, fuzzy rules 3.3.4, Bayesian methods 3.3.5, and causal methods 3.3.6. Each subsection provides a concise overview of the key aspects of these methods without delving into exhaustive detail to offer a comprehensive understanding of their approaches. The ultimate goal is to provide practical insights into how these methods work, how they are applied to RCA and their limitations.

3.3.1 Regression Models

Having explored various regression models in section 2.4.2.1, this part details its application in RCA for inference. We also present additional specific models.

3.3.1.1 Inference in Logistic regression

The primary objective of logistic regression is to determine the vector of coefficients β that best fits the training data. This process can be carried out using Maximum Likelihood Estimation (MLE), which aims to identify the parameter values that maximize the likelihood function. This allows us to compute the odds ratio, which is a key metric for interpreting the relative importance of each feature in predicting the binary outcome, i.e., the failure.

Definition 3.3.1 (Odds and Odds Ratio)**.** *Let $X^{(1)}, \ldots, X^{(p)}$ be the explanatory variables and Y the binary outcome. The **Odds** represents the probability of occurrence of the event relative to the probability of its non-occurrence. For the event, $Y = 1$ the odds are defined as:*

$$Odds(Y = 1) = \frac{P(Y = 1)}{P(Y = 0)} \tag{3.1}$$

*The **Odds Ratio** (OR) is a measure of association between an explanatory variable $X^{(k)}$ and the odds of the occurrence of an event Y. It quantifies the strength of the relationship between $X^{(k)}$ and Y by comparing the odds of Y occurring when the variable $X^{(k)}$ takes on a particular value $x^{(k)}$ to the odds of Y occurring when $X^{(k)}$ takes on a different value $\tilde{x}^{(k)}$.*

$$OR = \frac{Odds(Y = 1|\mathbf{X} = \tilde{\mathbf{x}})}{Odds(Y = 1|\mathbf{X} = \mathbf{x})} \tag{3.2}$$

where $\tilde{\mathbf{x}} = \left(x^{(1)}, \ldots, x^{(k-1)}, x^{(k)} + 1, x^{(k+1)}, \ldots, x^{(p)}\right)$.

Remark 3.3.2. *The logistic regression can be interpreted as a linear modeling of the "log odds" as the ratio $\frac{\pi}{1-\pi}$, where $\pi = P(Y = 1|\mathbf{X} = \mathbf{x})$, is the odds of the outcome.*

Consequently, we write the odds of $\mathbf{X} = \mathbf{x}$ as

$$Odds(Y = 1|\mathbf{X} = \mathbf{x}) = \frac{P(Y = 1|\mathbf{X} = \mathbf{x})}{P(Y = 0|\mathbf{X} = \mathbf{x})} = \exp(\beta^T \mathbf{x}) \tag{3.3}$$

and the Odds Ratio, as defined in equation 3.2,

$$\text{OR} = \frac{Odd(Y = 1|\mathbf{X} = \tilde{\mathbf{x}})}{Odd(Y = 1|\mathbf{X} = \mathbf{x})} = \exp(\beta^T(\tilde{\mathbf{x}} - \mathbf{x})) = \exp(\beta_k) \tag{3.4}$$

Hence, $\exp(\beta_k)$ is an estimate of the Odds Ratio which gives a measure of the association between the variable $X^{(k)}$ and the event.

3.3.1.2 Time Series approaches

The temporal aspect in RCA is often critical in identifying the underlying causes of failures. Regression methods tailored for analyzing data over time enable us to examine how variables change over time and assess their influence on the system. In the following, we describe two specific regression models: conditional logistic regression models and distributed lag models.

Conditional Logistic Regression (CLR) (Breslow & Day, 1980; Kuo et al., 2018) is a statistical modeling technique that builds upon logistic regression to analyze data structured as matched sets. These matched sets are frequently encountered in case-control studies. Discussed in the next chapter, they are observational studies comparing cases (individuals with the outcome of interest, e.g., a disease) and controls (individuals without the outcome) who share similar characteristics.

In fact, it uses the following elements:

- Stratification: Technique used to control confounding (factors that affect the analysis and prevent uncovering the true relationship, typically when exogenous variables influence both the dependent and target variables) by examining the association between exposure and outcome within different strata.
- Matching: Selection of unexposed subjects (controls) that have similar characteristics as the exposed subjects (cases) to address confounding and biases.

In the case-crossover design (Maclure, 1991), the strata represent time periods chosen from our data, and the matching is done on the same subject (self-matching).

This model is typically used to estimate an odds ratio and will model the odds of an outcome based on individual characteristics. The following study draws upon the methodology presented in Zhang et al. (1997) and has been applied to a case-crossover study. Let $\mathbf{x}_{i,t} = (x_{i,t}^{(1)}, \ldots, x_{i,t}^{(p)}) \in \mathbb{R}^p$ be the explanatory variable i.e the exposure for person i at a time step $t \in \{1, \ldots, T\}$ and let $y_{i,t} \in \{0, 1\}$ be a categorical variable indicating whether subject $i \in \{1, \ldots, N\}$ has the outcome. Given that odds are ratios, the actual model used is the logarithm of the odds, expressed as:

$$\log\left(\frac{\pi_{it}}{1-\pi_{it}}\right) = \lambda_i + \beta^\intercal \mathbf{x}_{i,t} \tag{3.5}$$

Where π_{it} indicates the probability of an event at time t, λ_i is the intercept and a constant of frailty associated to subject i, $\beta \in \mathbb{R}^p$ is the regression vector. Hence, the conditional probability of failure for subject i at time t is written as

$$\pi_{it} = \frac{\exp(\lambda_i + \beta^\intercal \mathbf{x}_{i,t})}{1 + \exp(\lambda_i + \beta^\intercal \mathbf{x}_{i,t})} \tag{3.6}$$

In the case-crossover approach, the exposures of cases denoted X_{i1} indexed by 1 are compared to the exposure of a reference control period denoted X_{i0} indexed by 0. For example, if the event occurs at 8 p.m., the control period could be at 8 a.m., a reference period taken far from the outcome and where there is no exposition. Note that the study can be done with multiple control periods instead of a single one. Note that the design allows for multiple control periods, but we assume that for each subject i, there is only one event.

Similarly to the logistic regression model, we developed how to write the likelihood function in appendix A. Consider $\mathbf{x}_{i,1} \in \mathbb{R}^p$ the vector of exposure for the case, $\mathbf{x}_{i,0} \in \mathbb{R}^p$ the exposure vector for the control and $y_{i,k} \in \{0, 1\}$ be the associated outcome result for $k \in \{0, 1\}$. Assuming that subjects $i \in \{1, \ldots, N\}$ are independent, the likelihood function is written :

$$L(\beta) = \prod_{i=1}^{N} \left[\frac{1}{1 + \exp\left(\beta^\top(\mathbf{x}_{i,0} - \mathbf{x}_{i,1})\right)}\right]$$

Inference By applying the maximum likelihood estimation method, we can estimate the parameters β using optimization techniques like Newton-Raphson. With the estimated parameters β, we can then calculate the Odds Ratio using an equation similar to equation 3.2 for the variable $X^{(j)}$:

$$\text{Odds Ratio} = \exp(\beta_j) \tag{3.7}$$

3.3.1.3 Distributed Lag Models

Principle Alongside autoregressive models, distributed lag models (Gasparrini & Leone, 2014; Judge et al., 1991; Schwartz, 2000) represent a class of dynamic models that incorporate the temporal dimension in order to explain a target variable. Autoregressive models are characterized by the dependence of the random variable to be explained, Y_t, on its past values. Distributed lag models, on the other hand, take a different perspective by asserting that observed variables are influenced by past values -**lags**- of explanatory variables represented by the random vector $\mathbf{X}_t = (X_t^{(1)}, \ldots, X_t^{(p)})$, implying a response time between the occurrence of external factors and its impact on the system.

In real life, the effect of an exposure may be spread over time or delayed, and may not be a simple direct cause-and-effect relationship. We must, therefore, define a model involving the previous occurrences of an exposure variable, i.e., lags with the future outcomes of the study.

For each multivariate time series, for $(\tau_1, \tau_2) \in \{1, \ldots, T\}^2$ such that $\tau_1 < \tau_2$, we denote $\mathbf{x}_{\tau_2:\tau_1} = (\mathbf{x}_t)_{t=\tau_1}^{\tau_2}$. Hence, for $\tau \in \{1, \ldots, T\}$, $\mathbf{x}_{\tau-1:1}$ contains the past values of $\mathbf{x}_\tau$ and $\mathbf{x}_{t:t-l+1}$ contains the l lags of $\mathbf{x}_{t+1}$ (for $l \geq 1$). The formulation is expressed as follows:

$$Y_t = f(\mathbf{X}_{t:t-l+1}, \ldots, Y_{t:t-l}, \epsilon_t)$$

where $\mathbf{X}_{t:t-l+1} = \{\mathbf{X}_t, \ldots, \mathbf{X}_{t-l+1}\}$, $l < t$ and ϵ_t is a Gaussian noise.

Definition 3.3.3. *A **Distributed Linear Lag Model** (DLM) describes a relation between an outcome Y_t and the explanatory variables X_t*

$$Y_t = \lambda + \sum_{k=0}^{l} \beta_k \mathbf{X}_{t-k} + \epsilon_t \tag{3.8}$$

where $\beta_k = (\beta_k^{(1)}, \ldots, \beta_k^{(p)}) \in \mathbb{R}^p$ for $k \in \{1, \ldots, l\}$ is a vector parameter of the model and ϵ_t is a centered white noise of variance σ^2 and such that $\forall t \in \mathbb{N}, \epsilon_t$ is independent from $(X_t, X_{t-1}, \ldots)$.

The coefficients β_k are called lag weights. They define the pattern of how $\mathbf{X}_{t-k}$ affects Y_t over time.

Equation 3.8 can be estimated by Ordinary Least Squares (OLS) when the error term is a white noise. One challenge with these models is collinearity, which can be mitigated by using non-linear lag models like polynomial lag models (Gasparrini et al., 2010).

A distributed lag model can be used in combination with Conditional logistic regression to analyze data where the effects of independent variables occur over time. The coefficients β_k would take into account a temporal aspect. Using the notations defined for equation 3.5 and considering $l \in \mathbb{N}$ to represent the number of lags, we can rewrite the equation as follows:

$$\log\left(\frac{\pi_{it}}{1-\pi_{it}}\right) = \sum_{j=1}^{p} \sum_{k=0}^{l} \beta_k^{(j)} x_{i,t-k}^{(j)} \tag{3.9}$$

where $x_{i,t-k}^{(j)}$ represents the j^{th} component of $x_{i,t-l}$, which is the j^{th} explanatory variable.

For instance, the overall effect of the variables on a single day is the impact on that day in addition to the impact on previous days. Estimating the coefficients $\beta = (\beta_1, \ldots, \beta_l) \in \mathbb{R}^{pl}$ becomes computationally challenging when the number of features p and the number of lag l are large. To address this, dimension reduction may be done by representing the coefficients in β as $\beta_k^{(j)} = \lambda_j w_k(\theta_j)$ using a specific pre-defined weight distribution w_k. Then, training process allows learning the lower-dimensional parameters $\theta = (\theta_1, \ldots, \theta_p)$ and $\lambda = (\lambda_1, \ldots, \lambda_p)$ in $\mathbb{R}^p$.

3.3.2 Association Rule Mining

Building on the introduction of association rule mining in section 2.4.2.2, this section details how rules are generated by incorporating additional constraints and explore the inference process.

3.3.2.1 Rule Generation

While support serves as a valuable indicator of item co-occurrence, it falls short of providing a comprehensive understanding of item relationships. To overcome this limitation, multiple metrics have been proposed in the literature that capture more information about the direction and strength of the association. In the following, we define a rule as an implication, $X \to Y$ where X and Y are itemsets from a set $\mathcal{I}$ drawn from a database $\mathcal{D}_b$. The rule is interpreted as "the antecedent X implies the consequent Y". Subsequently, we define some metrics used to generate such implication rules. For a more extensive review of metrics, we direct the reader to the following articles (Azevedo & Jorge, 2007; Lenca et al., 2007).

Definition 3.3.4 (Support of a rule)**.** *The support of a rule $X \to Y$ is the support (defined in 2.4.2) of the co-occurrence of itemset X and Y defined as :*

$$support(X \to Y) = support(X \cup Y)$$

where $\cup$ refers to the union of the sets.

Definition 3.3.5 (Confidence)**.** *The confidence of a rule $X \to Y$ where $X, Y \subseteq \mathcal{I}$ is defined as*

$$conf(X \to Y) = \frac{support(X \cup Y)}{support(X)} \tag{3.10}$$

Remark 3.3.6. *The **confidence** ranging within $[0, 1]$, is the percentage of transactions containing X that also contain Y. It is an estimate of the probability of observing Y given X, $P(Y|X)$, and is an indication of how often the rule has been found to be true. Consequently, confidence is a directed measure meaning that the confidence of the rules $X \to Y$ and $Y \to X$ can be different.*

While confidence measures the strength of an association rule, it does not distinguish between real and spurious associations that may arise by chance. The lift metric provides additional insights into a rule's significance.

Definition 3.3.7 (Lift)**.** *The lift of a rule $X \to Y$ where $X, Y \subseteq \mathcal{I}$ is defined as*

$$lift(X \to Y) = \frac{support(X \cup Y)}{support(X) \times support(Y)} \tag{3.11}$$

Remark 3.3.8. *The **lift** value, ranging from 0 to infinity, measures the strength of an association rule by comparing the observed support to the expected support if items X and Y were independent. A lift value close to 1 indicates that the items are independent, while a lift value significantly greater than 1 suggests a strong association and not a mere coincidence. However, lift is a symmetric measure and does not capture the direction of the association.*

The conviction metric complements confidence and lift by providing insights into the notion of implication in the rule.

Definition 3.3.9 (Conviction)**.** *The conviction of a rule $X \to Y$ where $X, Y \subseteq \mathcal{I}$ is defined as*

$$conv(X \to Y) = \frac{1 - support(Y)}{support(X \cup Y)} \tag{3.12}$$

Remark 3.3.10. *Conviction value, ranging from 0 to infinity, measures the degree to which the presence of an item in the consequent implies the presence of the item in the antecedent. Similar to lift, a high conviction value indicates a strong association.*

The following details the pseudocode that outlines the steps involved in the Apriori algorithm.

Pseudo Code (Agrawal & Srikant, 1994; Agrawal et al., 1993)

- Apriori's Candidate Generation

  The Generation function generates a candidate itemset using two steps: the candidate generation and the pruning step. The idea is to extend each frequent itemset of size k by adding other frequent itemsets. This process is fast and allows us to find every frequent itemset of size $k+1$. The pruning step is then necessary to avoid redundancy during the generation process.

Algorithm 1 Apriori_gen(L_k)

Require: Frequent itemset L_k of size k
Ensure: Candidate itemset C_{k+1} of size $k+1$
1: C_1 = all itemsets of size 1;
2: **for** (k=1; $L_k! = \emptyset; k++$) {
3: C_{k+1}= join L_k with itself; // Join step
4: **If** both $\{a_1, \ldots, a_{k-1}, a_k\}$ & $\{a_1, \ldots, a_{k-1}, a_{k+1}\}$ are in L_k {
5: add $\{a_1, \ldots, a_{k-1}, a_k, a_{k+1}\}$ to C_{k+1};
items are sorted
6: }
7: Remove k-itemsets in C_{k+1} that are not frequent; // Prune step
8: **return** C_{k+1};

- The Apriori Algorithm: Pseudo Code

  As described in the example above, the Apriori algorithm, using the frequent itemsets of size k, generates candidates C_{k+1} of size $k+1$. Then, it scans the database and calculates the support of each item. Finally, the algorithm selects all itemsets whose support satisfies the minimum support requirement and adds them to L_{k+1}.

Algorithm 2 Apriori

Require: transaction database $\mathcal{D}_b$, minimum support threshold Min_sup
Ensure: frequent itemsets
1: L_1= {{a}, a frequent item of size 1};
2: C_1= L_1;
3: **for** (k=1; $L_k! = \emptyset; k++$) {
4: C_{k+1}= Apriori_gen(L_k);
5: **for each** transaction T in database D do {
6: increment the count of all candidates in C_{k+1} that are included in T;
7: L_{k+1} = candidates in C_{k+1} with Min_sup
8: }
9: **return** $L = \bigcup_k L_k$;

Complexity The complexity of the *apriori* algorithm depends on several parameters, such as hyperparameters of the algorithm and data-dependent factors. This includes :

- Number of Items: The number of items in our data set directly affects the space complexity as the more items we have, the more space will be needed. We then need to store the support count of each item, a larger frequent set, and hence a larger number of candidate items.
- Support threshold: The number of frequent items is dependent on the support threshold. In fact, the more important this number, the more frequent itemset there will be.
- Number of transactions: Number of samples in the data set. This is directly increasing the complexity of the algorithm, as it is a recursive algorithm with several passes.
- Number of items in the rule: Depending on how complex we want the rule, we can choose the length of the output. If we choose a length $n \in \mathbb{N}$ then the algorithm will do n passes, increasing the complexity.

Time complexity can be calculated by separating different parts of the apriori algorithm:

- Generation of frequent itemset of size 1: This first step requires computing the support count of each item present in the transaction. Noting m the maximum length of the transactions, the operation requires $\mathcal{O}(Nm)$ where N is the number of transactions.
- Candidate generation: This step requires $\mathcal{O}(\sum_{k=2}^{m} k(k-2)|C_k|+|L_{k-1}|^2)$ for the generation and pruning.
- Support counting: The support count is done at each pass. Hence, the complexity is $\mathcal{O}(N \sum_{k=2}^{m} \binom{m}{k} \alpha_k)$ where m is the maximum length of the transactions and α_k is the cost for updating the support count.

More details on the calculations can be found in Tan et al. (2014).

3.3.2.2 Inference

Association rule mining algorithms use a variety of techniques to uncover hidden patterns and relationships within large data sets. The process typically involves three key steps:

1. **Data Pre-processing**: Raw data is transformed and organized to ensure compatibility with the association rule mining algorithm. For instance, continuous data, such as time series data, must be discretized using symbolic representation algorithms like Symbolic Aggregate approXimation (SAX) (Lin et al., 2007; Park & Jung, 2020) and other dimension reduction approaches presented in (Wang et al., 2010).
2. **Rule Generation**: Setting the parameters such as minimum support count and confidence levels and application of the algorithm to generate rules with a constraint on the consequence of the rule to be the "failure".
3. **Rule Evaluation**: Generated rules are sorted and evaluated using additional metrics such as confidence, lift, and conviction.

3.3.3 Decision Trees

Decision Trees, as described in section 2.4.2.2, are a popular choice in Root Cause Analysis for their strengths in interpretability and simple visualization.

3.3.3.1 Inference

To classify new data points, often drawn from a separate testing set, the decision tree algorithm traverses the tree structure based on the values of the input features. This process of navigating the tree stops at a leaf node, which represents a specific region in the feature space associated with a particular class label. The class label assigned to the new data point is the dominant class label among the data points reaching that leaf node.

Decision trees offer valuable insights for root cause analysis using classification or regression approaches. In classification, the goal is to predict the occurrence of a failure, and the objective is to construct a tree that effectively distinguishes between normal operation conditions and failure outcomes. The process typically involves two main steps:

1. **Decision tree learning**: Learn the tree structure from data using an algorithm such as CART.
2. **Analysis and Interpretation**: Analyze the decision tree to identify the root cause. The features that are near the top of the tree contribute most significantly to the outcome and can be potential root causes. In addition, features that are close to the leaves highly influence the outcome with specific decisions, which provides another level of granularity in root cause analysis.

Having established the effectiveness of decision rules in classifying root causes with ARM and decision trees, we now explore fuzzy logic to handle inherent uncertainties and vague descriptions of a failure by operators often present in RCA.

3.3.4 Fuzzy logic

Fuzzy logic developed by Zadeh (1965), allows taking into account a notion of uncertainty, imprecision and vagueness in a real-world scenario by extending the classical set theory (Crisp sets). It offers a framework for representing vague terms such as "low", "medium," or "high" using a rule-based system. In the following, we describe how fuzzy logic allows extracting imprecise relationships for RCA.

Definition 3.3.11 (Fuzzy Set and Membership function)**.** *Let $\mathcal{S}$ be a space. A **fuzzy set** F of $\mathcal{S}$ is characterized by a **membership function** $\mu_F : S \to [0, 1]$.*

Fuzzy sets differ from traditional sets in the fact that each element has a partial membership in a set which is measured through a membership function μ taking values in $[0, 1]$. The membership function could have different forms such as triangular, trapezoidal, and logistic functions (see figure 3.1 for an example). Note that for a crisp set, μ takes values in $\{0, 1\}$.

The membership function allows fuzzy systems to be defined in natural language using linguistic variables.

Definition 3.3.12 (Linguistic Variable)**.** *A **linguistic variable** corresponds to the triplet (V, R_V, F_V) where*

- *V is a variable*
- *S_V is the domain on which V is defined*
- *F_V is a finite or infinite collection of fuzzy sets*

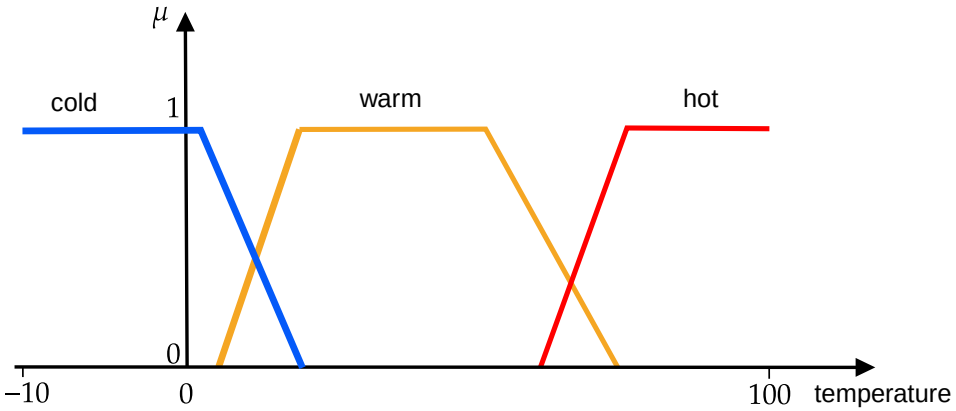


Figure 3.1: This figure represents the membership function of the fuzzy sets "cold","warm" and "hot" of the linguistic variable ("Temperature", [-10;100],("cold","warm","hot")).

As shown in Figure 3.1, if the variable V represents the temperature, R_V represent the temperatures values that V can take and F_V could include terms such as "cold", "warm" and "hot".

3.3.4.1 Fuzzy Operators

Fuzzy logic relies on a set of operators, such as AND, OR, and NOT, to manipulate imprecise information. In this part, we define these fuzzy logic operators, which differ from classical operators.

Definition 3.3.13 (Union (OR))**.** *For any element $x \in \mathcal{S}$ and, the membership function of the union of fuzzy sets A and B is*

$$\mu_{A \cup B}(x) = \max(\mu_A(x), \mu_B(x)) \tag{3.13}$$

The membership of the union captures the overall inclusiveness by taking the maximum.

Definition 3.3.14 (Intersection (AND))**.** *For any element $x \in \mathcal{S}$ and, the membership function of the intersection of fuzzy sets A and B is*

$$\mu_{A \cap B}(x) = \min(\mu_A(x), \mu_B(x)) \tag{3.14}$$

The membership of the intersection focuses on shared membership by taking the minimum.

Definition 3.3.15 (Complement (NOT))**.** *For any element $x \in \mathcal{S}$, the membership function of the complement of fuzzy set A is*

$$\mu_{\neg A}(x) = 1 - \mu_A(x) \tag{3.15}$$

3.3.4.2 Fuzzy Rules

While traditional rules rely on binary values, fuzzy rules provide a more flexible approach by incorporating uncertainty and ambiguity. This adaptability makes fuzzy rules particularly well-suited for real-world problems that involve imprecise or qualitative data by enabling the encoding of human knowledge and expertise. They are expressed as follows:

Definition 3.3.16 (Fuzzy Rule)**.** *Let A and B be fuzzy sets. For $x \in \mathcal{S}$, a fuzzy rule is in the form*

$$\textit{If } x \textit{ is } A \textit{ Then } x \textit{ is } B \tag{3.16}$$

Remark 3.3.17. *In the case of a rule "If x is A AND y is B Then z is C", the* ***fire strength*** $\mu = \min(\mu_A(x), \mu_B(y))$ *indicates the degree to which the rule matches the inputs.*

The **fire strength** quantifies the strength of the relationship between the IF and THEN of the rule. Each rule has an associated strength given by membership functions, indicating the degree of confidence in the rule.

Other types of rules exist such as certainty rules ("the more x is A, the more certain y lies in B"), gradual rules ("the more x is A, the more y is B" and m "the more x is A, the more possible B is a range for y". For additional details, see (Dubois & Prade, 1996).

3.3.4.3 Inference

Fuzzy inference systems use various inference mechanisms to process input data and generate output decisions. The process usually lies in the three following steps (Sabri, 2013):

1. **Fuzzification**: Convert the input value into a membership value of the fuzzy set.

2. **Fuzzy Inference**: Compute rules based on fuzzified inputs and evaluate them with fire strength.

3. **Defuzzification**: Transform the fuzzy output into a crisp.

The most widely used method is the Mamdani inference mechanism (Mamdani & Assilian, 1975), which ultimately converts the fuzzy output sets into crisp numerical values. The Sugeno method (Sugeno, 1985) differs from Mamdani by representing the output as a linear equation or constant value, simplifying the defuzzification process. Conversely, the Tsukamoto method (Saepullah & Wahono, 2015) incorporates a weighted average to combine the firing strengths of activated rules, offering a balance between the flexibility of Mamdani and the simplicity of Sugeno.

In the literature, a combination of fuzzy logic with association rules mining (Lin et al., 2010; Papadimitriou & Mavroudi, 2005) and decision trees (Yuan & Shaw, 1995; Zio et al., 2008) are used to extract meaningful rules from data.

Another alternative for reasoning under uncertainty is through Bayesian networks as introduced in section 2.4.2.3. These models capture the probabilistic relationships between variables. In the following, we detail the inference steps within this framework used in RCA.

3.3.5 Bayesian Networks

Bayesian Networks are commonly used in root cause analysis due to their ability to model and analyze probabilistic relationships between variables and their capacity to provide a visual representation of these relationships.

3.3.5.1 Inference

Root Cause Identification using BN After constructing a Bayesian network from data, the root cause of a failure can be inferred by analyzing the network structure and conditional probability distributions. The network structure reveals nodes connected to the problem variable, potentially indicating the root cause. The conditional probability distributions assess the relative importance of each potential cause in contributing to the problem.

Queries and Inference techniques The Bayesian network can be used to compute probability queries of interest, such as the dependence between a set of evidence variables corresponding to the failure Y, and the potential root cause X, expressed as $P_{\mathcal{G}}(X|Y)$. Two main families of inference methods are used to solve these queries (Lokrantz et al., 2018):

- **Exact inference algorithms**, such as Most Probable explanation (MPE) and marginal probabilities compute the posterior probabilities without approximations. However, exact inference becomes intractable for large and complex networks (Chan & Darwiche, 2012).
- **Approximate inference algorithms**, such as belief propagation and Monte-Carlo methods, provide a computationally efficient way to approximate the posterior probabilities. They make simplifying assumptions about the network structure and probability distributions to reduce the computational burden.

3.3.6 Causal discovery Methods

3.3.6.1 Causal Bayesian Networks

Although directed acyclic graphs (DAGs) can represent independence assumptions, they don't necessarily imply causation. In fact, they can represent any set of conditional independence relationships, regardless of the arrangement of the variables. The construction of a causal network allows for a more reliable and justified representation and also enables us to act and intervene appropriately rather than on the basis of spurious correlations.

A Causal Bayesian Network (CBN) (Pearl, 2000) is a Bayesian Network where directed edges between nodes represent causal relationships. Similarly to BN, it satisfies the causal Markov condition and hence can be factorized as in Equation 2.15. CBN's ability to address intervention queries, along with probabilistic queries, is one of its key strengths. In fact, this is the second level of causality according to Pearl & Mackenzie (2018), and makes it possible to respond to counterfactual answers. An intervention, denoted as $do(X = x)$, forces the value of X to be fixed at x, effectively altering the causal chain and allowing us to examine the impact of this artificial intervention. An intervention query takes the form of $P(Y|do(X = x))$, where Y represents the outcome variable and X represents the intervened variable. This query asks us to predict the probability of Y occurring if X is forced to take on the value x.

To define a CBN, let first define $P(V)$ as a probability distribution on a set $V = \{X^{(1)}, \ldots, X^{(p)}\}$ of random variables with a possible realization $v = \{x^{(1)}, \ldots, x^{(p)}\}$. Additionally, let $P(V = v|do(X = x))$ denote the interventional distribution, resulting from an intervention on a subset $X \subset V$ (including $P(V)$ with $X = \emptyset$ meaning that there is no intervention).

Definition 3.3.18 (Causal Bayesian Networks (Pearl, 2000))**.** *A DAG G is said to be a* ***Causal Bayesian Network*** *compatible with the set of all interventional distributions $P(V = v|do(X = x))$, with $X \subseteq V$, if, and only if:*

- *$P(V = v|do(X = x))$ is Markov relative to G for all v*
- *$P(x^{(k)}|do(X = x)) = 1$ with $X^{(k)} \subset X$ whenever $x^{(k)}$ is consistent with $X = x$ i.e the relation in the network is unchanged.*
- *$P(X^{(k)}|Parents(X^{(k)}), do(X = x)) = P(X^{(k)}|Parents(X^{(k)}))$ for all $X^{(k)} \not\subset X$ i.e $Parents(X^{(k)})$ is consistent with $X = x$.*

This definition enables us to compute the distribution resulting from any intervention $do(X = x)$ using a simpler structure, known as a truncated factorization.

$$P(V|do(X = x)) = \prod_{k|X^{(k)} \not\subset X} P(X^{(k)}|Parents(X^{(k)})) \tag{3.17}$$

Practically, intervening or applying the do operator in a variable X in the graph involves "mutilating" the network (Mahmood, 2021; Pearl, 2000) by removing incoming arrows to X. This modification facilitates the estimation of causal effects by analyzing the impact and focusing only on the fixed value of X and ignoring the usual dependencies.

In addition, causal Bayesian networks are typically learned using constraint-based algorithms. These algorithms rely on certain assumptions, which we discuss below.

Constraint-based algorithms To discover causal relationships from mere correlations, constraint-based algorithms rely on different assumptions, such as :

- i.i.d assumption: Some methods assume that the methods should take as input independent and identically distributed samples.
- Causal Sufficiency: the model assumes that the available variables are sufficient to uncover the causal relationship from the data. The model may assume that there are or are no unobserved (hidden) variables that influence the observed variables.
- Causal Markov condition: This condition states that the Markov condition holds for a causal graph. In other words, the statistical dependencies observed in the data can be explained by a causal graph, where the variables are conditionally independent of their non-descendants given their parents.
- Causal Faithfulness: this condition assumes that the observed statistical dependencies correspond to the true underlying causal relationship.
- Sample size: Some methods assume the availability of large samples to converge to the true causal graph.

Several causal discovery algorithms, including PC (Glymour et al., 2019; Spirtes et al., 2001) and GES (Spirtes et al., 2001), share similar assumptions: they require i.i.d. data, a large sample size, a causal Markov condition, and causal faithfulness. Under these conditions, both PC and GES are guaranteed to converge to the true Markov Equivalent Class. The FCI algorithm (Spirtes et al., 1993), while keeping a similar assumption, relaxes the constraint of causal sufficiency by allowing for the presence of hidden variables.

3.3.6.2 Structural Causal Models (SCM)

A Structural Causal Model (SCM) represents the causal relationships between cause and effect using functions. In a multivariate setting, each variable $X^{(i)}$ is expressed as:

$$X^{(k)} = f_k(Parents(X^{(k)}), \epsilon_k) \text{ for } k = 1, \ldots, p \tag{3.18}$$

where f_k is called the causal mechanism and is a deterministic function and ϵ_k is the noise independent of the set of parents Parents($X^{(k)}$). The aim is to find this equation that best matches the underlying data generation mechanism, hence allowing us to perform interventions. The objective is to learn the causal dependencies through the function f_k and take advantage of the asymmetry in the data or in the noise terms to determine the causal direction. In the following, we give an overview of the main methods developed:

- **Linear Non-Gaussian models** aim to identify causal relationships in linear models under the assumption of non-Gaussian noise. The main method is the Linear Non-Gaussian Acyclic Model (LINGAM) (Shimizu et al., 2006) which verifies three properties:
  1. The observed variables $X^{(k)}$ can be arranged in a causal order for $k \in \{1, \ldots, p\}$ noted σ_k, such that variable $X^{(\sigma(k+1))}$ cannot cause $X^{(\sigma(k))}$.

2. There is a linear relationship between a variable and the earlier variables in the causal order expressed as:

$$X^{(k)} = c_k + \sum_{j,\sigma(j)<\sigma(k)} \beta_{kj} X^{(j)} + \epsilon_k \tag{3.19}$$

where ϵ_k is a noise term associated with $X^{(k)}$ and c_k is a constant term.

3. The noise terms ϵ_k are continuous real-valued random variables with non-Gaussian distributions of non-zero variances. The ϵ_k are independent of each other i.e. the probability density function is of the form $P(\epsilon_1, \ldots, \epsilon_p) = \prod_k P(\epsilon_k)$

- **Non-Linear Methods**: These methods aim to model the non-linear relationships in the data-generating process.

  - **Non-Linear additive noise model** (Hoyer et al., 2008) assumes that a set of nonlinear functions can represent the causal relationships between variables and that the observed data is generated by adding noise to these functions. They are expressed as :

$$X^{(k)} = f_k(Parents(X^{(k)})) + \epsilon_k \text{ for } k = 1, \ldots, p \tag{3.20}$$

    where the noise terms ϵ_k are jointly independent and may have arbitrary probability densities.

  - **Post Non-Linear** (PNL) (Zhang & Hyvarinen, 2012; Zhang & Hyvärinen, 2010): These models take into account the possible sensor distortions in the observed variables through a non-linear function of the underlying mechanism. It is expressed as:

$$X^{(k)} = f_{k,2}(f_{k,1}(Parents(X^{(k)})) + \epsilon_k) \tag{3.21}$$

    where $f_{k,1}$ is a non-linear function denoting the non-linear effect of the causes, $f_{k,2}$ denotes a post-non-linear distortion in variable $X^{(k)}$ and ϵ_k is a noise term independent of $Parents(X^{(k)})$.

Note that the non-linear additive noise model is a special case of the PNL model where there is no distortion.

Traditionally, causal discovery methods have primarily focused on analyzing non-temporal data. Researchers then developed adaptations for these methods to handle time series data. We describe some of these adaptations in the next part.

3.3.6.3 Causal Discovery for Time Series

Existing causal discovery methods are primarily designed for independent and identically distributed data. Time series data, however, exhibits inherent dependencies between consecutive samples. These dependencies can be complex and non-stationary, making it challenging to apply traditional causal discovery methods directly. To address this issue, researchers have developed specialized definitions and methods tailored for time series data, which can effectively handle temporal dependencies and assess causal relationships in this dynamic context. The

addition of the temporal dimension has the effect of adding an assumption on the precedence associated to each node of the graph. Indeed, future elements cannot be the causes of past elements. In the following, we showcase a brief overview of the methods; for a complete survey, we direct the reader to (Assaad et al., 2022).

Granger Causality (Granger, 1969) is one of the earliest approaches to uncover causality from data. Particularly used in econometrics and time series analysis, the aim is to analyze regularities in the data and discover causal influence by determining whether a time series can predict another time series. We define Granger causality more formally as follows.

Definition 3.3.19 (Granger Causality (Granger, 1969))**.** *A time series, at time t, X_t Granger-causes Y_t if past values of X_t provide statistically significant information about future values of Y_t compared to using only past values of Y_t.*

Practically , considering two time series represented by X_t and Y_t, for $t \in \{1, \dots, T\}$, a first approach to assess if X_t Granger causes Y_t is to consider the following auto-regressive models:

$$Y_t = \beta_0 + \sum_{k=1}^{l} \beta_k Y_{t-k} + \epsilon_t \tag{3.22}$$

and with the past values of the second time series:

$$Y_t = \beta_0 + \sum_{k=1}^{l} \beta_k Y_{t-k} + \sum_{k=1}^{l} \alpha_k X_{t-k} + \tilde{\epsilon}_t. \tag{3.23}$$

where ϵ_t and $\tilde{\epsilon}_t$ are Gaussian noise.

The statistical tests are defined as

- Null Hypothesis (H_0): The past values of X_t do no provide significant information to predict Y_t.
- Alternate Hypothesis: The past values of X_t do provide significant information to predict Y_t.

To assess the validity of the null hypothesis, statistical tests are performed on the residuals. Common choices include the F-test based on the sum of squared residuals (SSR) or the Pearson chi-square test. The optimal lag length l can be estimated using information criteria such as AIC or BIC.

Granger causality methods utilize predictive information in time series to establish causal relationships, but this approach can produce misleading results as Granger causality does not necessarily imply causation. It primarily identifies correlations and requires additional assumptions to be considered for the relation to be causal, such as causal sufficiency and a sufficient sample size. Moreover, Granger causality was primarily developed in the bivariate and linear case.

To address some limitations, several extensions and improvements of the original Granger causality methods have been developed, including multivariate versions (Chen et al., 2004;

Geweke, 1982) and nonlinear variants like the nonlinear auto-regressive exogenous (NARX) model (Faes et al., 2008) and kernel-based methods (Marinazzo D., 2008).

Additionally, adaptations of the previously discussed models have been applied to time series data, including:

- **Graph based methods**: The methods discussed in section 2.4.2.3 and 3.3.5 have been adapted to handle temporal data. For example, the PC algorithm has been extended to time series data with the PCMCI algorithm (Runge et al., 2017), which uses a Momentary Conditional Independence (MCI) test to address autocorrelations in time series data. Additionally, to account for hidden variables, a time series version of the FCI algorithm, known as tsFCI (Entner & Hoyer, 2010), was developed by transforming the time series into a sample of random vectors with a sliding window. These approaches allow for the identification of conditional independence relationships between variables while effectively handling the temporal dependencies in the data.
- **Structural Causal models**: SCM have been extended to incorporate temporal dependencies, leading to approaches such as VAR (vector autoregression), VarLiNGAM (Hyvärinen et al., 2010), and Timino (Peters et al., 2013). These methods utilize noise-based models to infer causal relationships between variables while accounting for the temporal structure of the data.
- **Score-based**: Score-based algorithms have been developed specifically for time series data. One notable example is DYNOTEARS (Pamfil et al., 2020), which learns a dynamic Bayesian network to identify causal relationships in time-dependent systems of arbitrary order.

3.4 How our work fits in the litterature

The following chapter introduces a novel approach combining the case-crossover design, an epidemiological approach used to investigate acute disease triggers, and the Apriori algorithm. The resulting time series causal algorithm extracts rules of interest from a non-linear time series data set. In addition, a predictive rule-based algorithm demonstrates the potential of the proposed method.

The literature on interpretable models for root cause analysis encompasses a diverse range of techniques, including Bayesian Networks, regression models, decision trees, association rules mining, and fuzzy logic models. Decision rule-based models, such as decision trees and association rules, offer a valuable trade-off between interpretability and predictive accuracy. However, decision trees can be prone to overfitting and instability, while association rules may fail to capture causal relationships.

Random forests, introduced by Breiman (2001), have emerged as a popular alternative to decision trees, combining the strengths of multiple trees to reduce overfitting and improve predictive accuracy. However, the complexity of random forests, arising from the large number of trees and bagging technique, can hinder interpretability. More recent approaches, such as SIRUS (Bénard et al., 2021), have addressed the interpretability challenge by extracting interpretable rules from random forests. SIRUS identifies frequent patterns in the trees, enabling

the extraction of interpretable rules while maintaining the high accuracy and stability of random forests.

In contrast current interpretable association rule mining models for root cause analysis exhibit several limitations: they often fail to capture causal relationships, can be susceptible to instability and low accuracy, and may require significant adaptation to handle time series data effectively.

To address these limitations, the next chapter introduces a novel ARM approach that incorporates causality into a new framework. This method effectively addresses the shortcomings of traditional ARM models by extracting causal rules from time series data, providing more reliable and insightful root cause analysis.

CHAPTER 4

Rule-based Model

Contents

4.1 Introduction

4.1.1 General introduction

Monitoring has enabled, with the help of increased storage capacity, to collect a large amount of data. The data analysis plays a crucial role in understanding the underlying mechanisms and the occurrence of incidents. In the industrial context, this consists of placing sensors and collecting temporal data like temperature, flow rates, chemical characteristics, or wind power to capture the evolution and dynamics of the system. Exploiting these large amounts of temporal data is a real challenge facing many companies. Indeed, they contain enormous amounts of information that could help improve efficiency or optimize certain processes.

Driven by easy access to machine learning environments and the recent success of deep learning techniques, many models have been developed to predict the occurrence of these events, but they do not only work on their causes but also on the correlated variables. This makes these models less robust, as they could miss the incident by trusting a correlated variable. In areas where decisions and actions can have serious consequences, for example, on humans in medicine or on the profitability in the industry, it is necessary to understand the model's decisions and to carry out a causal study to act in a justified way. Hence, the objective of causality in an industrial context is to better understand the decisions made by artificial intelligence algorithms, find the causes of unexplained events, and develop maintenance policies that anticipate breakdowns. Therefore, a theoretical approach should be developed to provide a general framework that could work in an industrial environment. In particular, the approach should help the operators understand what are the mechanisms behind every decision that is taken and allow them to prevent the apparition of an incident by defusing its arrival.

The interest in causality is growing, and these studies are becoming essential in industry and in many other fields of applications. For instance, it is common for distillation units to have a recurrent problem, called flooding, occurring during petroleum refining. The causal study allows a better understanding of the origins of these problems and to develop a general approach that can be used on many systems such as wind turbines.

4.1.2 Outline

Our methodology, based on Granger causality analysis, employs a *case-crossover* design (Maclure, 1991) (developed in section 4.2) to investigate causal relationships in industrial data. It is an approach used in epidemiology in order to understand the origins of a phenomenon appearing suddenly (heart attack, accidents, injuries (Estberg et al., 1998; Maclure & Mittleman, 1997; Mittleman et al., 1993, 1995)). Establishing a causal relationship between an exposure and an event necessitates demonstrating that the occurrence of the exposure indeed causes the event. Identifying the causes of acute events is a complex challenge in epidemiology, and the rigorous analysis of data collected from patient cohorts plays a pivotal role in these investigations.

This design is combined with the association rule mining algorithm Apriori (Agrawal & Srikant, 1994) which aims at discovering relationships of interest between two or more variables stored in data sets. The advantage of this method is that it has a high interpretability (Toti et al., 2016), hence easier to understand for operators that could then be able to act and defuse the problem.

In the following sections, we introduce a comprehensive framework designed to discover the underlying causes of acute phenomena, i.e., that occur briefly over time. To showcase its effectiveness, we employ this framework to analyze a case study involving flooding events. We propose the Case-crossover APriori (**CAP**) algorithm, which provides association and causal rules explaining the occurrences of failures, and the Case-crossover APriori Predictive algorithms (**CAPP1** and **CAPP2**) that predict them. The purpose of this work is to answer the following research questions: What are the causes of the flooding event, and what are the variables involved in this phenomenon? This chapter is organized as follows. Section 4.2 is the presentation of the *case-crossover* design. Section 4.3 contains a depiction of the original methodology developed in this study. Section 4.4 is dedicated to the description of the flooding problem, the presentation of the data, and the results obtained from applying our approach. Section 4.5 contains the conclusive remarks and approaches for future work.

4.2 Case-crossover design

Finding the causes of acute events has always been a challenge for epidemiologists, and the way the data collected from a batch of patients is analyzed plays an important and crucial role in the study. A relation between a factor and an event is said to be **causal** if the occurrence of the factor causes the event.

To determine the root cause of a disease that manifests briefly among a population, it would be ideal to compare a healthy group of individuals to the identical group had they been exposed to some factors. However, this is not feasible in reality since we can only observe one of the groups, and the other group is a hypothetical situation called the **counterfactual**.

In practice, a widely employed approach involves comparing two distinct subgroups: one comprising individuals exposed to the event of interest and another consisting of those not exposed. To draw causal conclusions, we make an assumption that the outcome of the exposed individual represents the outcome that would have occurred had exposure not been present, they are exchangeable. This assumption is known as **exchangeability** (Greenland & Robins, 1986; Mittleman & Mostofsky, 2014).

This design called the *case-control* design, must fulfill some conditions in order to eliminate biases from the study. Specifically, it is crucial to control for **confounding**, which arises from inherent differences between the two groups. To minimize confounding, subjects in the different groups should share similar characteristics, such as age and gender. These biases can lead to spurious associations or mask genuine ones, and the *case-control* design is not specifically designed to prevent or control them.

4.2.1 Principle

The Case Crossover Design, proposed by Maclure (1991), is used in epidemiology to study the onset of acute events across a population and is widely used to find the causes of diseases. It is an alternative to the Case-Control design and allows for avoiding confusion and biases. This design has been widely employed in various research domains, including investigating the impact of air pollution on health (Jaakkola, 2003) and, more recently, exploring the potential causes of COVID-19 and colder temperatures in the increased mortality (Runkle et al., 2020).

The *case-control* design, as we have discussed, carries certain limitations stemming from the comparison of individuals with different characteristics. Without pre-processing our data and ensuring that similar subjects are matched, confounding factors may arise.

The *case-crossover* design addresses the limitations of the standard case-control approach by focusing exclusively on individuals who have experienced the event of interest. This design hinges on the comparison of two distinct periods: the control period and the case period. The control period is selected in a "normal" operating phase, often spanning a significant duration before the occurrence of the event of interest. The case period, on the other hand, is selected during the hazard period, a time frame preceding the event's onset. By comparing these two periods, we can identify the changes that occurred between them and determine which changes are consistently observed across the population and are likely to have played a role in triggering the event or disease. Moreover, this design eliminates the need for a separate control group, requiring only data from the individuals who experienced the event.

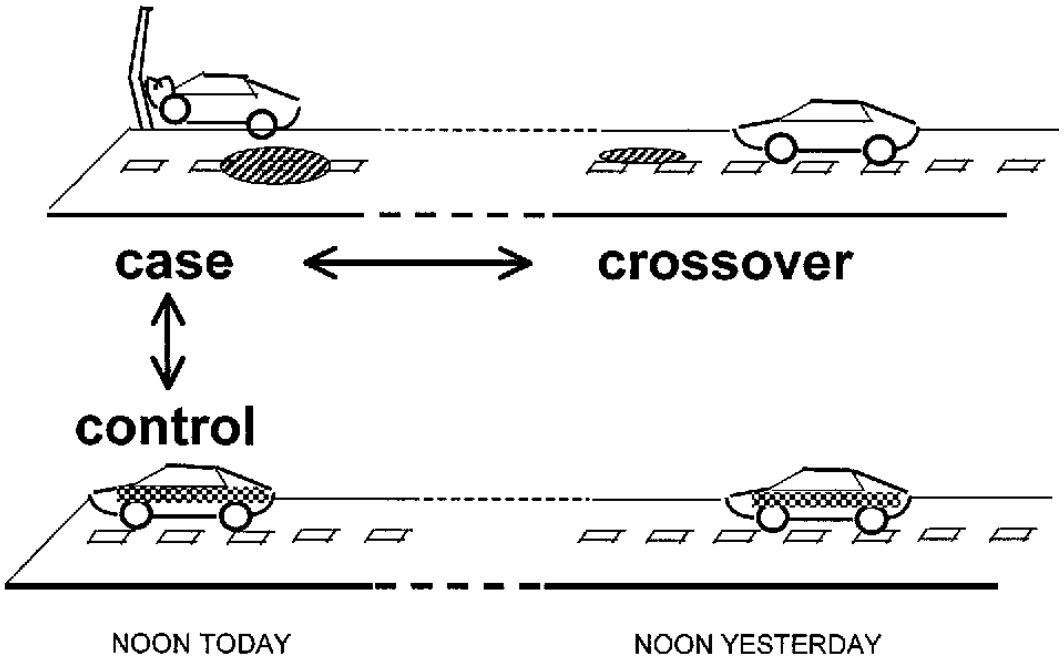


Figure 4.1: Description of the difference between case-control and case-crossover with a car accident example from Maclure & Mittleman (2000) article. The case-control design compares the impact of factors (e.g., driving while drung) of individuals who have experienced a car accident to a control group who did not experience it. The case-crossover design focuses on individuals who have experienced a car accident and analyzes the impact of factors during specific time periods, including the case period leading up to the accident and a control period prior to the accident.

Example 4.2.1 ((Maclure & Mittleman, 2000))**.** To illustrate the relevance of this design, consider the case of car accidents in Figure 4.1. To determine the root causes of accidents in a specific region, the *case-control* design involves identifying two groups of individuals: one that has experienced a car accident in the area – the case group – and another that has not – the control group. By comparing the behaviors of these two groups, we can identify any factors

that are more prevalent among individuals in the "case group" than in the "control group". For instance, we might discover that individuals in the control group are more likely to engage in behaviors such as eating or using their phones while driving, suggesting that these behaviors could be contributing factors to car accidents in the area.

Typically, this method involves monitoring several individuals on the same day of the week and at the same time over an extended period. Statistical analysis is then performed on the aggregated data to identify any significant associations between factors and the event.

In epidemiology, we note for each subject i, $\mathbf{x}_{i,t} \in \mathbb{R}^p$ the multidimensional vector of exposure covariates at time t where $t \in \{1, \dots, T\}$ is the time at which the event can occur and the outcome of the subject i at a time t is noted $y_{i,t} \in \{0, 1\}$ where

$$y_{i,t} = \begin{cases} 1 & \text{if the failure occurs} \\ 0 & \text{otherwise.} \end{cases} \tag{4.1}$$

In this study, the subject i is a time series where the flooding i happens for each $i \in \{1, \dots, N\}$ where $N \in \mathbb{N}$, $\mathbf{x}_{i,t}$ is the multidimensional vector of factors/covariates measured over time in the distillation unit and the event described by $y_{i,t} = 1$ is the flooding event happening at time t.

In the following, we outline some fundamental assumptions that the case-crossover design must satisfy to establish causal relationships. These assumptions ensure that the design effectively controls for confounding factors and provides reliable evidence.

4.2.2 Hypotheses

4.2.2.1 Acute event

An important assumption of the case-crossover design is that the causal factors leading to the event should be **acute/transient**. In other words, the event should be triggered by a brief but significant change in a variable during the hazard period, rather than a prolonged or cumulative effect. This assumption is crucial for eliminating bias and drawing causal conclusions. Indeed, if the exposure has a long-lasting effect, the case-crossover design will not be able to capture the temporal relationship between exposure and the event. For instance, if a driver had been drinking for an extended period before the accident, the case-crossover design would not be able to effectively distinguish the impact of alcohol consumption during the hazard period from its overall influence on the driver's behavior. In such cases, a case-control study would be more appropriate.

4.2.2.2 Exchangeability

In the *case-crossover* design, we aim to assess the influence of exposure, such as a factor variable exceeding a predefined threshold, on the occurrence of an event. In order to set a *case-crossover* design, we should verify the validity of the hypotheses of exchangeability between case and control periods. In their article, Mittleman & Mostofsky (2014) suppose the control of the following:

- Confounding variables: factors that influence both the exposure and the outcome. Their presence can distort the relationship between exposure and outcome, either by creating a spurious association between them or by hiding an existing one.
- Selection Bias: happens when the individuals and times in a study do not represent the entire population and time that should have been included. This can lead to distorted results, as the relationship between exposure and event may be different for the selected group than for the entire population. This concerns particularly control time selection biases that include:
  - Dependence with exposure: The selection of the control period is crucial, as it should represent a time when there is no exposition to the factor under investigation. Taking the example of the car accident, if the control period is too close to the event, the individual could have been using their phone, making it difficult to discern any difference between the control and case periods.
  - Multiple exposures: Exposures that occur once or repeatedly can create bias if only the first or last exposure is considered, and other exposure periods are ignored. This is because sampling from times when exposure is more or less likely can skew the estimate of the effect of the exposure on the outcome.
- Auto-correlation: occurs when the exposure in one time period is correlated with exposure levels in other time periods. They are of different types, including:
  - Auto-correlation between case and control: When the duration of the effect of an exposition is greater than the duration of the control period, this will lead to a bias. The previous example of the drunk subject, where the effects are present in both periods, shows the limitation of Auto-correlation.
  - Auto-correlation between outcomes: If the exposition weakens the subject, it will have an impact on the next observations. For example, if a runner has a knee injury, he will be more prone to get injured in the same area as he will be weakened.

To assure exchangeability between the time periods under comparison, it is crucial to address potential confounding variables, selection bias, and auto-correlation. If there is no exchangeability, the variable being studied may not be associated with the event or causal.

4.2.2.3 Existing applications

Case Crossover design have been created by M. Maclure to answer the specific question: Was this event triggered by something unusual that happened just before? The objective is to determine how we quantify the "before" and the "unusual" behavior.

The case-crossover design was initially employed in epidemiological studies, particularly for investigating the onset of Myocardial Infarction (MI). The approach aims to identify the factors contributing to MI by comparing the behavior of individuals during the onset period to their behavior during earlier healthy periods. This approach involved using the same individual as their own control by asking them about their activities during the previous days for comparison purposes. This allowed to reduce bias and to find and quantify many triggers of MI, such as physical exertion (Maclure, 1993), anger (Maclure, 1995), sexual activity (Muller et al., 1996),

cocaine use (Mittleman et al., 1999), bereavement (Mittleman et al., 1996) and respiratory infections (Meier et al., 1998).

Epidemiologists have also used the case-crossover design to investigate the factors that lead to injuries and examine the relationships between drug consumption and various phenomena. According to Maclure & Mittleman (2000), the case-crossover design has become known since the publication of the article (Maclure & Mittleman, 1997) in 1997, stating the association between mobile phone use while driving and car accidents. Since then, they have been used in several domains and for different studies with air pollution, such as its association with daily mortality (Lee & Schwartz, 1999) or injuries in racehorses (Estberg et al., 1998).

More generally, the Case Crossover design could be applied to multiple fields such as environment epidemiology (Laurent et al., 2007), pharma-epidemiology (Hebert et al., 2007), occupational health (Vegso et al., 2007) and economic health (Stevens et al., 2006). Although this design has a theoretical potential for non-epidemiological fields such as industrial environments, there have been very limited studies conducted in industrial settings using this approach.

4.3 Rule-based algorithm

Association Rule Mining (ARM) is a data mining framework that allows the extraction of frequent associations of variables in a database. It has the advantage of being highly interpretative and easy to understand. In this section, we describe how ARM has been used in retail and how we adapt it to more varied fields of application, particularly for time series, by introducing the CAP, CAPP1, and CAPP2 algorithms.

4.3.1 Motivation

ARM has been developed by Agrawal & Srikant (1994) for commercial purposes. Indeed, commercial enterprises accumulate a significant amount of data on a daily basis. In the case of supermarkets, consumer purchases that can be retrieved from checkout receipts are a huge source of information. Their analysis helps to better understand consumers' behavior and thus establish appropriate marketing campaigns, better manage inventories, or improve customer relations.

The general setting for ARM is composed of a database containing transactions, and each transaction is an item-set, i.e., a set of items. Let I be the set of items and D be the set of transactions, which is a set of item-sets of I, and let a be an item-set and b an item. Rules extracted are of the form $a \rightarrow b$. Several challenges can arise when employing association rules. The number of generated rules can become overwhelming, especially for large data sets, making it impractical to examine all possible associations. In fact, in a database with n items, the number of rules of the form $a \rightarrow b$ for all possible item-sets a and items b that are not present in a is $n2^{n-1}$, hence the complexity would be exponential and the problem intractable. In addition, we may find rules with random patterns that do not actually reflect any real connection or cause-and-effect relationship between the items.

ARM algorithms allow finding relationships between items from the database in the form of association rules which are rules (implications) of the form $a \rightarrow b$ where a is an item-set and b is an item-set that is not present in a. In our case, we consider that b is only one item. In this section, we use the Apriori algorithm to extract causal rules from a database. Given our

objective of identifying causal relationships, we deem Apriori's capabilities as sufficient for our purpose. Alternative algorithms, such as FP-Growth, could be explored in future endeavors.

4.3.2 Notations

In the following, we focus on the supervised binary classification framework using the *case-crossover* design. In the section 4.4, we show that an appropriate preprocessing of the data set allows the selection of control and case periods on the same time series ("individual"). For each multivariate time series $(\mathbf{x}_t)_{t=1}^T = (\mathbf{x}_1^\intercal, \dots, \mathbf{x}_T^\intercal)^\intercal \in \mathbb{R}^{p\times T}$ and for each $(\tau_1, \tau_2) \in \{1, \dots, T\}^2$ such that $\tau_1 < \tau_2$, we denote $\mathbf{x}_{\tau_1:\tau_2} = (\mathbf{x}_t)_{t=\tau_1}^{\tau_2}$. Hence, for $\tau \in \{1, \dots, T\}$, $\mathbf{x}_{\tau-1:1}$ contains the past values of $\mathbf{x}_\tau$ and $\mathbf{x}_{t:t-l+1}$ contains the l lags of $\mathbf{x}_{t+1}$ (for $l \geq 1$).
Suppose, that we have a sample composed of N pairs $\mathcal{D}_N = \{(\mathbf{x}_{i,T:1}, y_i), i = 1, \dots, N\}$ where the N pairs are i.i.d of the same law as $(\mathbf{x}_{T:1}, y)$. Each time series $\mathbf{x}_{i,T:1}$ is generated by the same stationary process, and $y \in \{0, 1\}$ is the binary outcome. For a time series $(\mathbf{x}_t)_{t=1}^T$, the goal is to predict the binary output y. Hence, the objective is to find an interpretable and causal predictive model of the event $y = 1$ given $(\mathbf{x}_t)_{t=1}^T$.

In this section, we propose an original method inspired by the *case-control* and the *case-crossover* designs which can process continuous or categorical temporal data. The method aims at finding an interpretable and causal predictive model of the event $y = 1$ given $(\mathbf{x}_{[1,\delta]}, \mathbf{x}_{[T-\delta+1,T]})$ where δ is the duration of a period and $\Delta = T - 2\delta + 1$ is the gap between the two periods, as shown in Figures 4.3 and 4.4. First, using prior and domain knowledge, we select the periods that allow us to characterize the event. Secondly, we need to transform the periods into a categorical data set. Since the association rule mining algorithm only works with categorical data, we should indeed apply a transformation to convert continuous variables into categorical data without losing relevant information. This allows to extract simple rules explaining the dynamics of the phenomenon.

4.3.3 Methodology

We decided to apply the *case-crossover* design on a data set using association rule mining by creating an algorithm called Case-crossover APriori (**CAP**). The first step is to set up an environment in which we are able to compute rules. We need to define what our "transactions" are and the type of "items" that will be included in our rules. As rules are computed between periods of the time series, we need to set a metric that creates the items.

4.3.3.1 Case-crossover Design

Basic design Firstly, the *case-crossover* design needs to be adapted to a time series data set designed for classification. It is done by constructing a parametric model that could be optimized, allowing the user to select the control and case periods and fine-tune the model by selecting the best parameters.

In time series data, case samples are instances where the target variable indicates a failure ($y_i = 1$), while control samples represent instances where the target variable indicates no failure ($y_i = 0$). This approach allows us to use a machine learning framework for binary classification, whether we want to compute interpretable rules or make predictions.

The primary goal is to identify appropriate control periods. Due to the limited number of failures in the data set, preprocessing is crucial for employing the Apriori algorithm. The case-crossover design suggests comparing a control period with a case period. This approach is illustrated in the Figure 4.2.

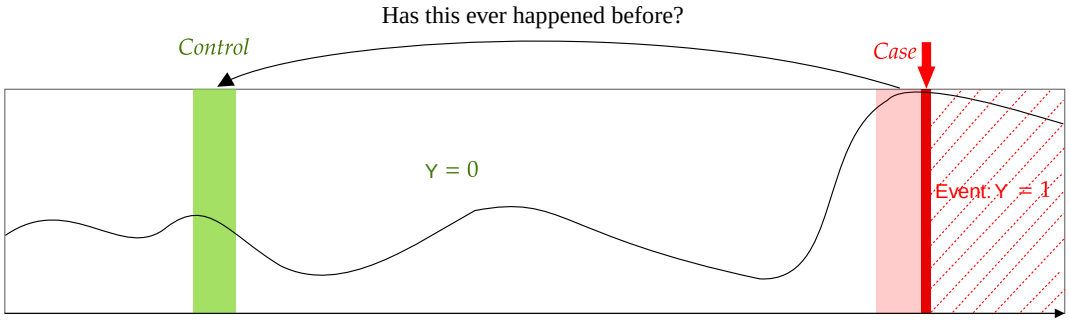


Figure 4.2: Basic case-crossover design with a single control and case period.

However, this strategy presents a limitation for our study as it only considers pairs of data where the case period precedes a failure, essentially a pair leading to a failure. To effectively conduct association rule mining and machine learning in general, we need to investigate every outcome, including cases that do not lead to failures. This necessitates identifying control periods for events that do not result in failures.

Adapting Case-crossover design To adapt the case crossover design to our binary classification framework, we require samples with both failure and non-failure labels for association rule mining, we introduced an additional period, as depicted in Figure 4.3 and 4.4.

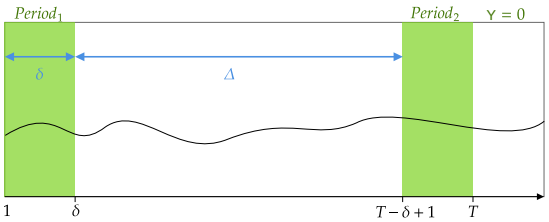


Figure 4.3: Proposed design applied for a time series without the failure: **control**

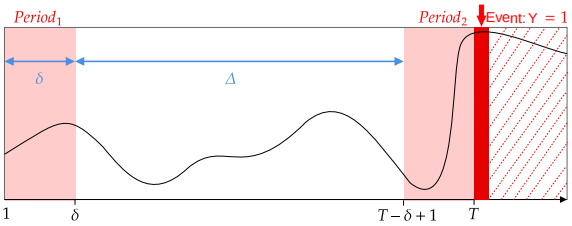


Figure 4.4: Proposed design applied for a time series with the failure: **case**

In this design, the algorithm compares period_1 and period_2 that have the same duration δ (which is a hyper-parameter) and that are separated by an interval of duration $\Delta = T - 2\delta$. Hence, for each time series $(\mathbf{x}_t)_{t=1}^{T}$ taken in $\mathcal{D}_N$, we can extract the first period $\mathbf{x}_{\text{period}_1} = (\mathbf{x}_t)_{t=1}^{\delta}$ and the second period $\mathbf{x}_{\text{period}_2} = (\mathbf{x}_t)_{t=T-\delta+1}^{T}$.

This approach allows us to compute rules or to make predictions by using a machine learning framework for binary classification.

4.3.3.2 From continuous data to items

We experimented with various methods for mapping the data for each period and each input variable into a single value (discretization) while preserving the system's dynamics. We identified three approaches:

- **Autoregressive Model** The objective is to detect a loss of stationarity as it could be a cause of failure using autoregressive models. An autoregressive model is a statistical model that predicts the current value of a time series based on a linear combination of its past values. Let $(z_1, \ldots, z_t)$ be a time series, an autoregressive model of order $p \in \mathbb{N}$, $AR(p)$ can be written as:

$$z_t = \beta_1 z_{t-1} + \beta_2 z_{t-2} + \ldots + +\beta_p z_{t-p} + \epsilon_t$$

  where $\beta_i \in \mathbb{R}$ and ϵ_t is the realization of a white noise process of variance σ^2. Two autoregressive models will be fitted on a period of data taken one hour before the given period, one for the control and the other for the case period. Then, using the estimated parameters of the model, we will predict the evolution of the hour following respectively the control and the case period. Finally, the residual error for each given period will be computed.

- **Standard deviation** For each selected period, we calculate the standard deviation for each variable.

- **Mean** For each selected period, we computed the mean for each variable.

Ultimately, we found that computing the mean of each selected period for each variable provided the most convenient and relevant approach. This transformation is easy to understand and interpret, and it allows us to construct counterfactual scenarios.

Absolute values of a period In the basic version of the case-crossover design as shown in Figure 4.2 and in the adapted version in Figure 4.4 and 4.3, a first approach deals with the most recent period only i.e. $\bar{x}_{\text{period}_2} = (\bar{x}^{(j)}_{\text{period}_2})_{j=1}^{p}$ summing up the dynamics (absolute value) of the system of the case and control period.

If we take the example of Figure 4.4, we select the period on the right (the most recent) and compute the mean of the period. In Figure 4.4, we observe a significantly higher mean value before the event compared to the mean value observed in Figure 4.3 which can indicate an influence of the variable.

Change between pairs: How can we compare a pair of periods? In the adapted version of the case-crossover design as shown in Figure 4.4 and 4.3, once we have the values $\bar{x}_{\text{period}_1} = (\bar{x}^{(j)}_{\text{period}_1})_{j=1}^{p}$ and $\bar{x}_{\text{period}_2} = (\bar{x}^{(j)}_{\text{period}_2})_{j=1}^{p}$ summing up the dynamics of the system during the first and second periods, we need to compare them. The metric chosen is the **percentage change**:

$$\left(\bar{x}^{(j)}_{\text{period}_1}, \bar{x}^{(j)}_{\text{period}_2}\right) :\rightarrow \left| \frac{Max(\bar{x}^{(j)}_{\text{period}_1}, \bar{x}^{(j)}_{\text{period}_2}) - Min(\bar{x}^{(j)}_{\text{period}_1}, \bar{x}^{(j)}_{\text{period}_2})}{Max(\bar{x}^{(j)}_{\text{period}_1}, \bar{x}^{(j)}_{\text{period}_2})} \right| \tag{4.2}$$

If we take the example of Figure 4.4, we first select the two periods shown in red in the figure. Then, we compute the mean of each period. Finally, we compute the percentage change of the means. In Figure 4.4, there is a large increase in the mean value between the two periods, hence a large value of the percentage change, while in Figure 4.3 there is no meaningful change between the two periods.

Let us denote by f_{abs} the function taking as input one period $\mathbf{x}_{\text{period}_2}$ and computing its mean and f_{pairs} the function taking as input the first and the second period $\mathbf{x}_{\text{period}_1}$ and $\mathbf{x}_{\text{period}_2}$, computing the mean of each period $\bar{x}_{\text{period}_1}$ and $\bar{x}_{\text{period}_2}$ and additionally compute the percentage change using the metric (4.2). Finally, the problem is formulated as follows: the objective is to find a predictive model of the event

$$y = 1 \text{ given } f_{abs}(\mathbf{x}_{\text{period}_2}) \text{ and } f_{pairs}(\mathbf{x}_{\text{period}_1}, \mathbf{x}_{\text{period}_2})$$

Categorization step Association rule mining algorithms, like the Apriori algorithm, take as input categorical variables. Hence, we need to do a "categorization step" because when we have either absolute values, or we compare the pair $\bar{x}^{(j)}_{\text{period}_1}$ and $\bar{x}^{(j)}_{\text{period}_2}$ using the percentage change metric (4.2). Note that values from the percentage metric lie between 0 and 1 if we consider that all variables take positive values (this hypothesis is not restrictive for continuous real-valued random variables as it is always possible to transform them into random variables with uniform distribution over [0,1]). Then, we categorize these values into two categories for absolute values **DOWN** and **UP** and three for percentage changes **LOW**, **MEDIUM**, and **HIGH**. This was decided for clarity reasons to explain the method, but it can be extended to an arbitrary number of categories.

Let us note for each $j \in \{1, \dots, p\}$, $\mathbf{x}^{(j)}_{\{\text{down}\}}$ the boolean variable which is True if the mean is below the median value and $\mathbf{x}^{(j)}_{\{\text{up}\}}$ if above. Moreover, let us note for each $j \in \{1, \dots, p\}$ $\mathbf{x}^{(j)}_{\{\alpha,\beta\}}$ the boolean variable which is True if the percentage change of $\mathbf{x}^{(j)} = (\mathbf{x}^{(j)}_{T:1})$ falls in the interval defined by the quantiles of order α and β. We first estimate two empirical quantiles from the data set, the quantiles of order 0.33 and 0.66. Thus, for each $j \in \{1, \dots, p\}$, we have three boolean variables to indicate the range in which the percentage change is: $\mathbf{x}^{(j)}_{\{0,0.33\}}$, $\mathbf{x}^{(j)}_{\{0.33,0.66\}}$ and $\mathbf{x}^{(j)}_{\{0.66,1\}}$.

For each of the time series $(\mathbf{x}_{i,T:1})$ taken in $\mathcal{D}_N$, we select period_1 and period_2, compute their means and compare them using the percentage change metric defined in (4.2). After completing the process for all the samples in the database $\mathcal{D}_N$, we set up the Table 4.1.

In the "Event ID" column, for $i \in \{1, \dots, N\}$ y_i is the outcome of the time step $T + 1$. If the event happens, then $y_i = 1$, otherwise $y_i = 0$. Hence, the comparison between the case and control allows for identifying variations characteristic of the event and separating them from independent variations. The "Items" column gathers, for each of the "Event ID", the interval in which the absolute value and the percentage change of each variable is.

Then, we perform one-hot encoding (Pedregosa et al., 2011) to transform categorical variables into numerical representations, as shown in table 4.2. Additionally, we create a boolean "Event" column by converting the "Event ID" column into a binary indicator of the occurrence of the event. Finally, we apply the Apriori algorithm to identify rules that predict the occurrence of an event.

Event ID	Items
y_1	$(\mathbf{x}_1^{(1)})_{\{0,0.33\}} = False$,$(\mathbf{x}_1^{(1)})_{\{0.33,0.66\}} = False$, $(\mathbf{x}_1^{(1)})_{\{0.66,1\}} = True$,$(\mathbf{x}_1^{(2)})_{\{0,0.33\}} = True$, $(\mathbf{x}_1^{(2)})_{\{0.33,0.66\}} = False$,$(\mathbf{x}_1^{(2)})_{\{up\}} = False$,...
y_2	...,$(\mathbf{x}_2^{(3)})_{\{0.33,0.66\}} = True$,..., $(\mathbf{x}_2^{(5)})_{\{0,0.33\}} = True$,$(\mathbf{x}_2^{(5)})_{\{0.33,0.66\}} = False$, ...
y_3	$(\mathbf{x}_3^{(1)})_{\{0,0.33\}} = False$, $(\mathbf{x}_3^{(1)})_{[0.33,66]} = False$, $(\mathbf{x}_3^{(1)})_{\{0.66,1\}} = True$,...
...	..., ...
y_{n-2}	...,$(\mathbf{x}_{n-2}^{(2)})_{\{0,0.33\}} = True$,..., $(\mathbf{x}_{n-2}^{(4)})_{\{down\}} = True$,...
y_{n-1}	...,$(\mathbf{x}_{n-1}^{(2)})_{\{0,0.33\}} = True$,...,$(\mathbf{x}_{n-1}^{(4)})_{\{0.33,0.66\}} = True$
y_n	...,$(\mathbf{x}_n^{(1)})_{\{0.33,0.66\}} = False$,$(\mathbf{x}_n^{(1)})_{\{0.66,1\}} = True$, ...,$(\mathbf{x}_n^{(3)})_{\{up\}} = True$,...

Table 4.1: Table constructed from the comparisons of the selected periods

Event ID	$(\mathbf{x}^{(1)})_{\{0,0.33\}}$	$(\mathbf{x}^{(1)})_{\{0.33,0.66\}}$	$(\mathbf{x}^{(1)})_{\{0.66,1\}}$	$(\mathbf{x}^{(2)})_{\{0,0.33\}}$	$(\mathbf{x}^{(2)})_{\{0.33,0.66\}}$	...
y_1	0	0	1	1	0	...
y_2	0	1	0	0	0	...
y_3	0	0	1	0	0	...
y_4	1	0	0	0	1	...
y_5	0	1	0	1	0	...
⋮	⋮	⋮	⋮	⋮	⋮	⋮

Table 4.2: One-hot encoding

4.3.3.3 Apriori algorithm

We used the Apriori algorithm from the package MLxtend (Raschka, 2018). The library provides parameters to customize the search for rules. The minimum support threshold **min_support** dictates the minimum number of occurrences for an itemset to be considered as a **frequent item-sets** 2.4.4. Additionally, the user can fine-tune the thresholds for metrics like **confidence** 3.3.5, **lift** 3.3.7, and **conviction** 3.3.9 to further discriminate between rules. The maximum length **max_len** parameter controls the maximum number of items and associations in the extracted rules. Finally, by adding a constraint to have only rules which have a target "Event=True" i.e. $y = 1$, and a max_len of 1, the CAP algorithm could find rules like: Finally, we impose a constraint to restrict the extracted rules to those with a target value "Event=True," effectively capturing patterns associated with failures i.e. $y = 1$, and with a max_len of 1, the CAP algorithm could find rules like:

$$\{\mathbf{x}_{\{0,0.33\}}^{(1)} = True\ \} \implies \{\text{Failure=True}\}$$

4.3.4 Predictive Algorithm

Beyond the evaluation of the rules found by the Apriori algorithm which is made by experts, we want to test predictive properties by creating a first predictive algorithm that we call Case-crossover APriori Predictive 1 (**CAPP1**). The goal is to predict the binary output y based on simple and understandable rules. We selected the first 10 rules by order of confidence and lift to do the prediction on a test time series X of length T.

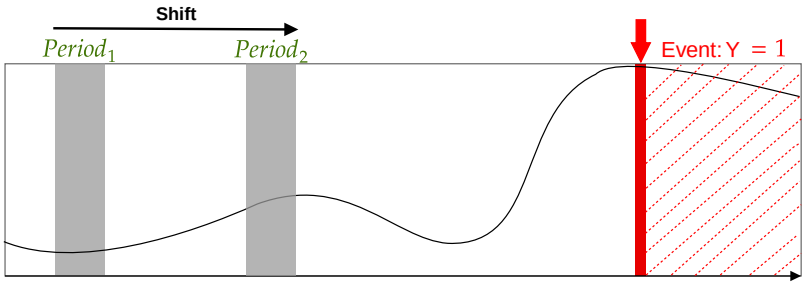


Figure 4.5: The gray pair of periods indicates a period to be evaluated, and it is shifted over time to identify potential failures when the rules are triggered.

In Figure 4.5, we compute the antecedent of an aggregation of rules that have been found. If at least one rule is True, CAPP1 predicts a "failure" and triggers the alarm. We have experimented several other approaches to perform rule-based prediction, among them, we can cite the simple aggregation technique which consists of voting on the found rules similar to what the SIRUS algorithm does (Bénard et al., 2021). The aggregation could be improved by using an ensemble learning method such as stacking (Hastie et al., 2009) by learning the decision combining these rules. The perfect predictive algorithm would predict a "failure" for Figure 4.4 but not for Figure 4.3.

Example To better understand the process, let us take the example using the first rule of Table 4.3.

$$\{\mathbf{x}^{(1)}_{\{0,0.33\}} = \text{True}, \mathbf{x}^{(2)}_{\{0.33,0.66\}} = \text{False}\} \implies \{\text{Failure} = \text{True}\} \tag{4.3}$$

We consider a test time series $(\mathbf{x}_t)_{t=0}^{T}$ and select the first and second variables $X^{(1)}$ and $X^{(2)}$ and compute their percentage changes between period_1 and period_2. If the percentage change of the variable $X^{(1)}$ is less than the quantile 0.33 and that of the variable $X^{(2)}$ is not between the quantiles 0.33 and 0.66, the algorithm predicts an event.

In order to estimate the error of our predictive model, we need to classify the predictions into four outcomes: the True Positive (TP), the True Negative (TN), the False Positive (FP), and the False Negative (FN). Then, we use the following metrics:

- True Positive Rate (TPR) or Recall summarizes the fraction of examples assigned to the positive class that belongs to the positive class

$$TPR = \frac{TP}{TP + FN}$$

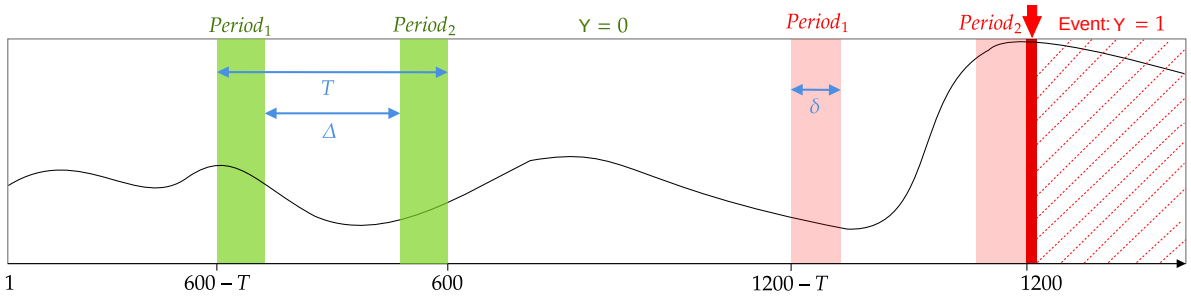


Figure 4.6: This figure shows how, from one time series of duration 20 hours (1200 minutes), we make a cutout to obtain the control sample in green and the case sample in red.

- Similarly, True Negative Rate (TNR) summarizes how well the negative class is predicted

$$TNR = \frac{TN}{TN + FP}$$

- F2-score is a weighted F-score and is used when it is much worse to miss a True Positive than to give a False Positive

$$F2 = (1 + 2^2) \times \frac{precision \times recall}{(2^2 \times precision) + recall}$$

These metrics can be used on the training database, on a test set, and in cross-validation.

Case-crossover APriori Predictive 2 (CAPP2) A complementary approach called Case-crossover APriori Predictive 2 (CAPP2) has been studied in order to improve the quality of prediction. Indeed, in addition to looking for the rules leading to an event of the form

$$\{\mathbf{x}^{(1)}_{\{0,0.33\}} = True, \mathbf{x}^{(2)}_{\{0.66,1\}} = True\} \implies \{\text{Event} = \text{True}\}$$

we have also looked for the contraposed, which are the rules that do not lead to an event (leading to "Event = False") of the form:

$$\{\mathbf{x}^{(4)}_{\{0.33,0.66\}} = True, \mathbf{x}^{(3)}_{\{0.33,0.66\}} = True\ \} \implies \{\text{Event} = \text{False}\}$$

Let us call "Event=True rules" the first rules and "Event=False rules" the second ones. There are different ways to combine these two approaches to compute a more robust predictive model. We could adjust the number of rules proving to be True for each of the two approaches, give more weight to the "Event=True rules" for the prediction, or give more weight to the "Event=False rules." Cross-validation allows testing, observing, and studying the behavior of each of these experiments. The decision could be improved by learning the decision combining the two types of rules.

4.4 Application

This section provides a comprehensive overview of our algorithm's application. We begin by introducing the real-world problem that we aim to address, followed by a detailed description of the data employed for the analysis. Finally, we present the obtained results, showcasing the effectiveness of our algorithm.

4.4.1 Flooding

Petroleum refining is a complex process that transforms crude oil into various usable products, including gasoline, diesel, and feedstocks for petrochemicals. The first step in this process is distillation as shown in figure 4.7 and 4.8, a method that separates different liquid substances based on their boiling points. In the context of petroleum refining, distillation is crucial for separating the hydrocarbon fractions present in crude oil.

Regularly, an event called flooding (KISTER, 1990; Ludwig et al., 2009; Oeing et al., 2021; Peiravan et al., 2020) occurs and requires the process to be stopped for a considerable amount of time. This happens when the steam flow is too high and blocks the flow of liquid in the column in figure 4.8. They are usually detected by sharp increases in differential pressure and a decrease in production performance. This is a frequent problem and the exact cause can vary: excessive vapor flow, too much heating, etc. Flooding results in a loss of performance and a decrease in the quality of separation. Conventional methods based on theoretical equations and/or temperature and differential pressure analysis have been used to develop predictors. These attempts have so far been unsatisfactory; either the number of false positives was too high, or a large number of flooding was missed.

Flooding is a common problem in distillation columns that can disrupt the separation process and lead to significant production losses (Oeing et al., 2021; Peiravan et al., 2020). It occurs when the vapor flow exceeds the capacity of the column, causing the liquid to back up and accumulate on the trays. This can be detected by sudden increases in differential pressure and a decline in the column's separation efficiency. Several factors can contribute to flooding, including excessive vapor flow, excessive heating, or improper column design. Flooding results in reduced separation efficiency and, as a result, a decrease in product quality and financial losses.

Conventional methods for predicting flooding, such as those based on theoretical equations or temperature and differential pressure analysis, have proven unsatisfactory due to high false positive rates or missed flooding events.

Flooding events are complex, and their causes are not fully understood, but we know that there are different types of flooding with varying origins. Currently, a predictive model has been developed to anticipate the occurrence of flooding events. The Random Forest algorithm (Breiman, 2001) enables one-hour advance warnings. While the model has high accuracy, it still generates false positives and fails to detect some events. These shortcomings can lead to wasted time and financial losses.

One significant drawback of this approach is that the Random Forest algorithm is a black-box model, which means that even though it can forecast flooding occurrences, the underlying causes of the prediction of these events remain opaque. Indeed, Random Forest is an ensemble of decision trees, where each tree makes a prediction based on a different subset of data, and the

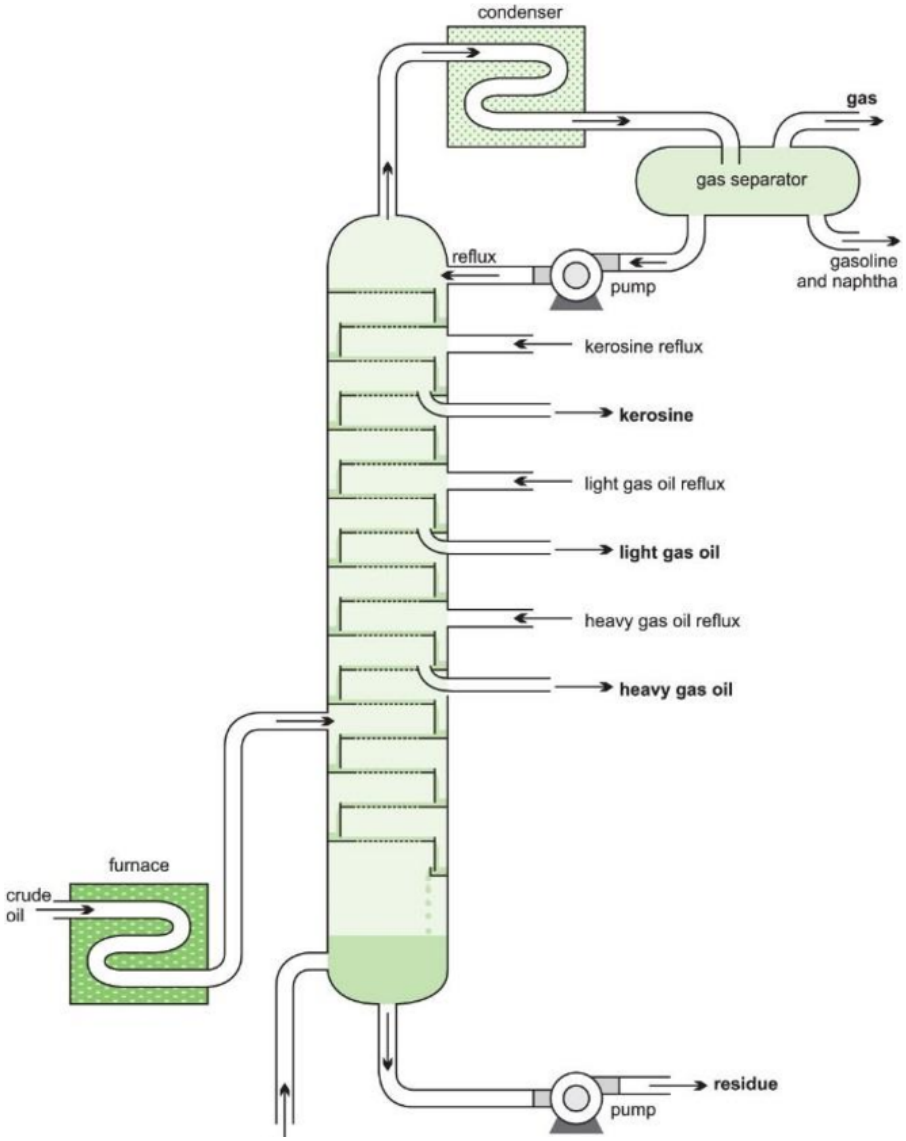


Figure 4.7: Distillation unit (Montanus, 2016)

final prediction is determined by aggregating the votes among the trees. In addition, Random Forest is not a causal model, which means it cannot establish direct relationships between variables, potentially compromising the reliability of its predictions After implementing the predictive model on-site for real-time flooding detection, operators raised several questions: How does the algorithm make its predictions? Why should they trust a model that they do not understand and cannot be explained? What actions should be taken to prevent flooding when the alarm is triggered? These questions have remained unanswered, and we believe it is crucial to address them.

A causal study is thus necessary to develop an interpretable model that extracts the relationship between the variables and the onset of the flooding. Based on expert insights, we assume exchangeability, including that each flooding is an acute event and that they are isolated incident and independent from other occurrences. Hence, the case period will be selected just

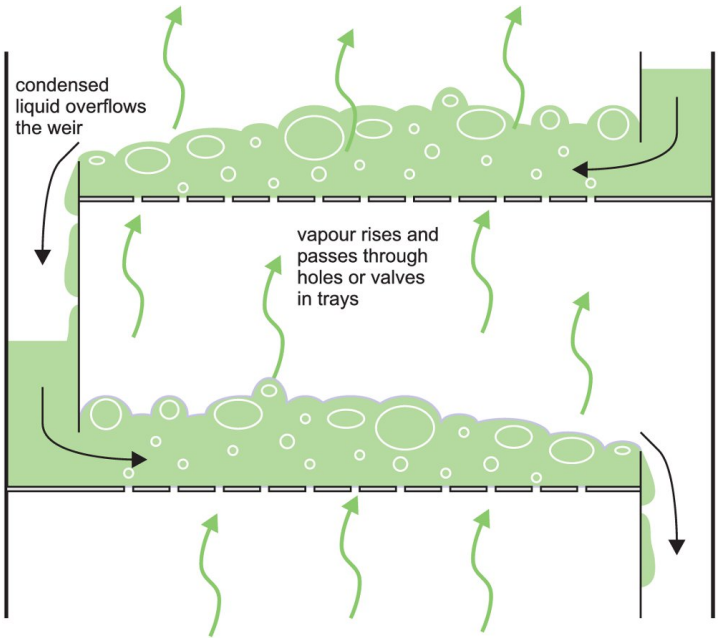


Figure 4.8: This figure shows the movement of liquid components downward while vapor rises inside the distillation column (Lichtarowicz, 2016).

before the event and the control period should be far enough from the case and in normal operating conditions of the distillation column.

4.4.2 Data

Numerous sensors were placed at various points in the distillation unit to collect data and monitor the evolution of the system. More than 800 variables were measured, providing information such as the type of input crude, pressures, temperatures, flow rates, valve openings, and chemical measurements. The variables are categorical or continuous and take positive values. These measurements were carried out every minute for 4 months, and the identification of flooding events is calculated using a formula involving variables from the outputs of the distillation column and is presented in the data in the form of an additional binary column where a 0 represents a normal operation condition system and 1 represents the flooding event. Each column represents a measured variable, and each line describes the system at a specific minute.

In our study, we consider that flooding events are independent of each other. For this reason, we only take into account events that occur at least 20 hours apart from each other. Thus, we identify a total of $N = 38$ long time series (of duration 20 hours=1200 minutes) to be studied. We have, therefore, cut the data into 38 long time series, the last moments of which correspond to the appearance of the flooding event. In order to build the database $\mathcal{D}_N$ using the *case-crossover* design, we must have pairs $(\mathbf{x}_{T:1}, y)$. Therefore, we need to define the duration T of the time series and get samples such that we have couples with labels $y = 1$ and $y = 0$ coming from the same long time series. The label $y = 1$ is simple to obtain because we just have to select the last moments of each of the 38 time series, which, by definition, all end with a flooding event. For the label $y = 0$, we had to sample and select a part of the 38 series. Since we

rules	support	confidence	lift
$\{\mathbf{x}^{(1)}_{\{0,0.33\}} = True, \mathbf{x}^{(2)}_{\{0.33,0.66\}} = False\} \implies \{\text{Event=True}\}$	0.2315789	0.9777778	1.955556
$\{\mathbf{x}^{(1)}_{\{0,0.33\}} = True, \mathbf{x}^{(3)}_{\{0.33,0.66\}} = False\} \implies \{\text{Event=True}\}$	0.2236842	0.9770115	1.954023
$\{\mathbf{x}^{(1)}_{\{0,0.33\}} = True, \mathbf{x}^{(4)}_{\{0,0.33\}} = False\} \implies \{\text{Event=True}\}$	0.2210526	0.9767442	1.953488
$\{\mathbf{x}^{(5)}_{\{0,0.33\}} = True, \mathbf{x}^{(2)}_{\{0.33,0.66\}} = False\} \implies \{\text{Event=True}\}$	0.3118421	0.9753086	1.950617
$\{\mathbf{x}^{(1)}_{\{0,0.33\}} = True, \mathbf{x}^{(6)}_{\{0.33,0.66\}} = False\} \implies \{\text{Event=True}\}$	0.2052632	0.9750000	1.950000
$\{\mathbf{x}^{(5)}_{\{0,0.33\}} = True, \mathbf{x}^{(2)}_{\{0.66,1\}} = False\} \implies \{\text{Event=True}\}$	0.2552632	0.9748744	1.949749
$\{\mathbf{x}^{(1)}_{\{0,0.33\}} = True, \mathbf{x}^{(7)}_{\{0.33,0.66\}} = False\} \implies \{\text{Event=True}\}$	0.2013158	0.9745223	1.949045
$\{\mathbf{x}^{(5)}_{\{0,0.33\}} = True, \mathbf{x}^{(3)}_{\{0,0.33\}} = False\} \implies \{\text{Event=True}\}$	0.2355263	0.9728261	1.945652
$\{\mathbf{x}^{(1)}_{\{0,0.33\}} = True, \mathbf{x}^{(2)}_{\{0,0.33\}} = False\} \implies \{\text{Event=True}\}$	0.2315789	0.9723757	1.944751
$\{\mathbf{x}^{(1)}_{\{0,0.33\}} = True, \mathbf{x}^{(8)}_{\{0.33,0.66\}} = False\} \implies \{\text{Event=True}\}$	0.2315789	0.9723757	1.944751

Table 4.3: This table displays the rules found by the algorithm, sorted by confidence and lift. The support is also shown here.

assume that the samples are independent, we have to select this period so that it is far enough from the flooding event and under normal operating conditions. With the advice of experts, we decided to select samples at a time distance of 10 hours=600 minutes from the flooding event. This step of selection of periods requires preliminary knowledge of the phenomenon in order to select the periods of "normal" and "abnormal" functioning. In our case, we know that the event is acute and occurs in the hour before the event.

Figure 4.6 summarizes the principle of the *case-crossover* design and highlights the data cutout to obtain the control and case of Figure 4.3 and Figure 4.4.

Therefore, to learn rules, we have a training database $\mathcal{D}_N = \{((\mathbf{x}_{i,T:1}), y_i), i = 1, \ldots, n\}$ where $n = 76$. 38 samples of $\mathcal{D}_N$ have a label $y_i = 1$ and 38 samples have a label $y_i = 0$. The sampling is done every minute, and we have 4 hours of measurements for each sample, hence $T = 240$.

4.4.3 Interpretable Rules found by CAP

In this subsection, we use expert knowledge of the characteristic times of important phenomena to determine certain time parameters, such as δ. For the rest of the parameters, we did not want to optimize them too much to avoid overfitting, optimizing the thresholds is an idea to keep in mind if the learning base is large enough.

After preprocessing the data, we computed the Apriori algorithm with the described design with a period duration of $\delta = 60$, 1 hour sampled every minute, and a gap $\Delta = 120$ of 2 hours between period$_1$ and period$_2$. We set $min\_support \geq 0.2$ and $min\_len = 2$ and sort the results by confidence and lift. The rules that have been found are shown in Table 4.3.

Among the rules, we can see the presence of $X^{(1)}$ which is a variable computed from a physical model and used to be, before the random forest model, the variable allowing to determine the appearances of flooding events. Moreover, $X^{(2)}$ is a re-circulation flow variable and has been selected by experts as being very likely to explain the flooding appearance.

4.4.4 CAPP1 Prediction Results

To prevent overfitting and evaluate well the CAPP1 method performance, we decided to do a Leave-Two-Out (LTO). For $j \in \{1, ..., n/2\}$, we take the $(2j-1)^{th}$ and $(2j)^{th}$ element of the database $\mathcal{D}_N$ for testing, such that we have a couple computed from one of the 38 long time series with one element having a label $y = 0$ and the other $y = 1$, and we take the $n-2$ other elements of $\mathcal{D}_N$ as a training set. The training set provides data to the Apriori algorithm in order to learn rules using the different metrics we defined. The rules are then sorted by confidence and lift and are ready to be tested. For the testing, as described in subsection 4.3.4, we predict the two elements in the test set. Finally, we evaluate the prediction by computing the True Positive Rate, the True Negative Rate, and the F2 score and compute the mean of these scores over the 38 tests we have done with our LTO. Thus, in the following, all calculated scores are obtained by cross-validation.

As mentioned in section 4.3.4, we select the 10 rules with the highest confidence and lift and with two or fewer explanatory variables, then we calculate the quality of the prediction using the defined metrics. We evaluate the predictive performance of the CAPP1 method by a comparison with the one of a random forest (RF) algorithm. The RF is trained with the data set D_n and takes as input the averages of the input variables over $[T-\delta+1, T], [T-2\delta, T-\delta+1], \ldots, [1, \delta]$ and predicts the binary label "there is a flooding at time $T+1$ minute". The results are shown in Table 4.4.

We could always increase the True Positive Rate by choosing a higher threshold for the minimum support and increasing the number of rules, but this will directly affect the True Negative Rate as there is a trade-off between True Positives and False Positives. If our model is more sensitive and often rings an alarm, it will make more errors and then more False Positives.

The results are satisfactory as the True Positive Rate is relatively high and far better than a random prediction without even optimizing our algorithm but is insufficient compared to the random forest algorithm.

4.4.5 CAPP2 Prediction Results

After several tests, we opted for the following combination: we set min_support ≥ 0.01 and sorted the results by confidence with a minimum threshold of 0.5. If at least one out of the first 100 "Event=True rules" and less than one out of the first 100 "Event=False rules" is True, we predict that the tested pair leads to a flooding event i.e. $y = 1$. Otherwise, we predict that the pair does not lead to a flooding event, i.e., $y = 0$. Since a minimum threshold of confidence has been set, the number of rules can be smaller but limited to 100. Note that the choice of 100 rules here is empirical and depends on the choice of the minimum support threshold.

CAPP2 has allowed us to improve our prediction results and obtain the scores presented in Table 4.4.

Algorithm	F2 score	TNR	TPR/recall
Random Forest	0.8127	0.8368	0.8684
CAPP1	0.6991	0.6644	0.8684
CAPP2	**0.9139**	0.9210	0.8947

Table 4.4: Prediction scores.

These results are promising as the CAPP2 method achieves better scores than RF without optimizing our model with a relatively small data set and especially with a model that proposes a causal analysis.

4.5 Conclusion And future works

We have developed a data-driven model based on the *case-crossover* design and association rule mining for determining the causes of an incident from time series. This approach overcomes two main issues: the lack of interpretability and prediction based on correlations. The understanding of incidents is essential because it would allow predicting in advance their appearance using a causal prediction algorithm and be able to justify the reliability and confidence contrarily to a black-box algorithm.

The application and study of this approach to our data set provide conclusive results, confirming that the method is promising. This work gives insight to operators working in the refinery with the distillation unit and allows them to understand the mechanisms that trigger the event. The method finds interesting rules and describes associations between variables leading to an event. Among the top rules sorted by confidence, we find the variables that have been suspected to be causal by the experts. The associations make it possible to strengthen them and to add missing information necessary to the understanding of the phenomenon of flooding. In addition, our predictive study has shown that we could build a strong predictive model that could outperform the one actually in production. Indeed, the results of the four-month data set have confirmed these expectations, and there is still a lot of room for improvement.

This method selects certain parameters using expert knowledge. In the absence of such information, methods to determine these characteristic times must be considered, and more failure case data may be needed for this.

Several approaches have been identified for future work. Among them, we could cite the following ideas: instead of choosing two arbitrary quantiles as we did in this work, we could optimize them and adapt their number. We could also deepen the contraposed approach CAPP2, improve our predictive model by aggregating the results over multiple analyses with different parameters Δ and δ, and optimize the event detection system.

CHAPTER 5

Dynamic Modeling in Multivariate Time Series

Contents

This chapter explores dynamic modeling and begins by introducing the challenge of modeling systems that evolve over time in section 5.1. This naturally connects to the field of time series forecasting, for which an overview is given in section 5.2. Section 5.3 formally defines the problem of dynamic modeling, formulating the process of uncovering underlying models using forecasting techniques. Then, section 5.4 provides a review of the current state-of-the-art interpretable methods within dynamic models. Finally, section 5.5 outlines the specific challenges we address and presents our research question.

Throughout this chapter, let N, H, and $p \in \mathbb{N}$ denote respectively the number of time series of the training data set, the prediction horizon, and the dimension, i.e., the number of variables. For each multivariate time series $(\mathbf{x}_t)_{t=1}^{T} \in \mathbb{R}^{p \times T}$, $(\tau_1, \tau_2) \in \{1, \dots, T\}^2$ such that $\tau_1 < \tau_2$, we denote $\mathbf{x}_{\tau_2:\tau_1} = (\mathbf{x}_t)_{t=\tau_1}^{\tau_2}$. Hence, for $\tau \in \{1, \dots, T\}$, $\mathbf{x}_{\tau-1:1}$ contains the past values of $\mathbf{x}_\tau$ and $\mathbf{x}_{t:t-l+1}$ contains the l lags of $\mathbf{x}_{t+1}$ (for $l \geq 1$).

5.1 Introduction

Dynamic modeling aims to describe how systems change over time by identifying the mathematical equations that govern their behavior (Džeroski & Todorovski, 1993; Koza, 1994). These equations reveal the relationships between the system's variables, allowing us to understand the underlying dynamics of sequential data and predict future evolution. This approach is applied in various domains to gain deeper insights into the mechanisms at play, including physics, finance, biology, and climate science. There are two main types of dynamic models (Both, 2021):

- **Differential Equations** describe how variables change continuously over time through:
  - **Ordinary Differential Equations (ODEs)**: describe the evolution of one or several variables e.g $\frac{d\mathbf{x}_t}{dt} = \mathbf{f}(\mathbf{x}_t, t)$ where $\mathbf{x}_t \in \mathbb{R}^p$ represents the state variable at time t.
  - **Partial Differential Equations (PDEs)**: describe the evolution of a spatially-dependent process e.g $\frac{\partial \mathbf{y}_t}{\partial t} = \mathbf{F}(\mathbf{y}_t, \nabla \mathbf{y}_t, t)$ where ∇ is the spatial gradient and $\mathbf{y}_t$ is the spatially dependent process at time t. They can handle highly complex systems but require significant computational resources to find solutions.
- **Discrete-Time Models** describe the evolution of variables at discrete time intervals in two ways (Camps-Valls et al., 2023):
  - **Explicit**: describe the state of the system at the next step from current or previous steps with dependent variables e.g $\mathbf{x}_{t+1} = f(\mathbf{x}_t, \dots, \mathbf{x}_1)$ with the particular case where $\mathbf{x}_{t+1} = f(\mathbf{x}_t)$ and with a specific number of lags l, $\mathbf{x}_{t+1} = f(\mathbf{x}_t, \dots, \mathbf{x}_{t-l+1})$.
  - **Implicit**: describe the process of describing the system $\mathbf{x}_{t+1} = f(\mathbf{x}_t, \mathbf{z}_{t+1})$ through hidden structure $\mathbf{z}_{t+1} = g(\mathbf{z}_t)$ within data. Techniques such as dimensionality reduction (e.g., PCA) and transfer operator (e.g., mode analysis) are used.

The following section deals with explicit discrete-time models. Our goal is to learn the underlying dynamics directly from observational time series data without any prior knowledge of the system. This presents several challenges, including the need to capture the system's behavior with interpretable models to deal with non-linear dynamics and noisy data.

We first present a quick introduction to time series forecasting to establish the foundation. Then, we explore interpretable methods used for dynamic modeling in the literature. Due to the inherent complexities of time series modeling, the following analysis assumes a stationary data-generating process.

5.2 A quick overview on time series forecasting

The field of time series forecasting offers a rich variety of methods, with three main techniques dominating the landscape: linear regression models, tree-based models, and deep learning models. Since interpretability is the focus of our thesis, this section considers some of the common regression models and key strategies used in time series forecasting.

5.2.1 Introduction

The task of multivariate time series forecasting aims to predict future values of one or multiple target variables of interest by identifying patterns in historical values of inter-related time series variables. The forecast can be in the form of a point forecast, a prediction interval, a percentile, or a distribution (Petropoulos et al., 2022). In this chapter, we focus on point forecasts.

Multivariate forecasting models learn relationships between the variables from a data set by extracting patterns and implicitly uncovering underlying dynamics under some constraints. Despite a lack of consensus and classification of forecasting models (Januschowski et al., 2022; Petropoulos et al., 2022), forecasting models are often categorized into two categories: "statistical models" and "machine learning models". Barker (2020) distinguishes machine learning models, which are unstructured -making no assumptions about the data's underlying process - like decision trees and Deep Neural Networks, from statistical models, which are structured with well-defined assumptions such as autoregressive models. In addition, several other categories of analysis levels exist, as described by Januschowski et al. (2022), such as data-driven or model-driven and interpretable or predictive models.

Furthermore, beyond the type of model, the size of the horizon is another important aspect of time series forecasting (Bontempi et al., 2013). While forecasting at one-step ahead is challenging, multi-step forecasting introduces further complexities like reduced accuracy due to factors such as error propagation and increased uncertainty.

The next sections explore forecasting models, with a particular focus on regression analysis. We then cover the key time series forecasting strategies for both one-step and multi-step predictions.

5.2.2 Time Series Regression models

This section explores regression models used in time series forecasting, assuming a linear relationship between the target variable and the explanatory variables. In this setting, the explanatory variable is also called the regressors, independent or explanatory variables, while the target variable is also referred to as the regressand, dependent or forecast variable (Hyndman & Athanasopoulos, 2018).

Consider N multivariate time series $(\mathbf{x}_{i,T:1}) = (x^{(1)}_{i,T:1}, \ldots, x^{(p)}_{i,T:1})$, $i \in \{1, \ldots, N\}$, generated by the same stationary process and where explanatory variables at time t are represented by the

random vector $\mathbf{X}_t = (X_t^{(1)}, \ldots, X_t^{(p)})$. The time series can be written in vector form $\mathbb{R}^{pT}$ as :

$$\mathbf{x}_{T:1} = \begin{bmatrix} \mathbf{x}_T \\ \mathbf{x}_{T-1} \\ \vdots \\ \mathbf{x}_2 \\ \mathbf{x}_1 \end{bmatrix}. \tag{5.1}$$

In each time series, we extract pairs of sequences $\mathbf{z}_{i,t} = ((\mathbf{x}_{i,t:t-l+1}), (\mathbf{x}_{i,t+H:t+1}))$ for $t \in \{l, \ldots, T-H\}$. All the $\mathbf{z}_{i,t}$ have the same distribution $\mathcal{D}$ and we denote by $\mathcal{D}_N = \{\mathbf{z}_{i,t} : i = \{1, \ldots, N\}, t = \{l, \ldots, T-H\}\}$ the set of samples identically distributed with this law.

5.2.2.1 Autoregressive models

Autoregressive models, also called AR process (Hyndman & Athanasopoulos, 2018), are a class of linear models used for time series analysis and forecasting. These models, represented by a function f, use past observation of a single variable of interest to predict its future value at a given time. The AR process is written using a one-dimensional time series:

$$X_t^{(1)} = f(X_{t-1}^{(1)}, X_{t-2}^{(1)}, \ldots, X_{t-l}^{(1)}) + U_t^{(1)} \tag{5.2}$$

where $X_t^{(1)}$ is the one-dimensional variable to forecast, l is the lag order and $X_{t-j}^{(1)}$ is the predictor variable for $j \in \{1, \ldots, l\}$. $U_t^{(1)}$ is the error term, which captures the unexplained variance in the model that can arise from measurement error, unobserved variables, or limitations inherent to the model structure. A more general form of equation 5.2 includes additional lagged values of the explanatory variables.

Consider the linear autoregressive process for illustration, given as follows:

$$X_t^{(1)} = \beta_0 + \beta_1 X_{t-1}^{(1)} + \beta_2 X_{t-2}^{(1)} + \ldots + \beta_l X_{t-l}^{(1)} + U_t^{(1)} \tag{5.3}$$

They are useful as they are interpretable and provide a clear understanding of the relationships. Their primary strength lies in their interpretability, offering a clear understanding of the relationships between past and future observations within a time series.

5.2.2.2 Vector Autoregressive models

Vector Autoregressive models (VAR) extend the univariate autoregressive model to handle multivariate time series. VAR is widely used in time series forecasting (Lütkepohl, 2013; Zivot & Wang, 2006), as they capture the interdependencies between multiple variables, considering the past values of all variables in the system to predict the future values of each variable simultaneously. When dealing with l lags, the VAR(l) model is expressed as:

$$\mathbf{X}_t = \mathbf{c}_0 + C_1 \mathbf{X}_{t-1} + \ldots + C_l \mathbf{X}_{t-l} + \mathbf{U}_t \tag{5.4}$$

where $\mathbf{c}_0 \in \mathbb{R}^p$ is a vector of intercept, $C_k \in \mathbb{R}^{p \times p}$ is a coefficient matrix for $k \in \{1, \ldots, l\}$ and $\mathbf{U}_t$ is a p-dimensional zero-mean white noise process with positive definite covariance matrix $\mathbb{E}[\mathbf{U}_t \mathbf{U}_t^\intercal] = \Sigma_U$.

The VAR(l) model shown in equation 5.4 can be rewritten compactly as a VAR(1) model as follows:

$$\underline{\mathbf{X}}_{t:t-l+1} = \mathcal{C}\underline{\mathbf{X}}_{t-1:t-l} + \underline{\mathbf{U}}_{t:t-l+1} \tag{5.5}$$

where $\underline{\mathbf{X}}_{t:t-l+1}$ and $\underline{\mathbf{U}}_{t:t-l+1}$ are random vectors taking values in $\mathbb{R}^{pl+1}$, and $\mathcal{C} \in \mathbb{R}^{(pl+1)\times(pl+1)}$ is known as the companion form of the VAR(l). They are defined as:

$$\underline{\mathbf{X}}_{t:t-l+1} = \begin{bmatrix} 1 \\ \mathbf{X}_t \\ \mathbf{X}_{t-1} \\ \mathbf{X}_{t-2} \\ \vdots \\ \mathbf{X}_{t-l+1} \end{bmatrix}, \quad \mathcal{C} = \begin{bmatrix} 1 & 0 & \dots & 0 & 0 \\ C_1 & C_2 & \dots & C_{l-1} & C_l \\ I_p & 0 & \dots & 0 & 0 \\ 0 & I_p & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & I_p & 0 \end{bmatrix}, \quad \underline{\mathbf{U}}_{t:t-l+1} = \begin{bmatrix} 1 \\ \mathbf{U}_t \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{5.6}$$

where $I_p \in \mathbb{R}^{p\times p}$ is the identity matrix of size $p \times p$.

In the following, we denote $\underline{\mathbf{x}}_{t:t-l+1}$ and $\underline{\mathbf{u}}_{t:t-l+1}$ the vector realizations for the random vectors $\underline{\mathbf{X}}_{t:t-l+1}$ and $\underline{\mathbf{U}}_{t:t-l+1}$ respectively.

Example 5.2.1. To illustrate, consider the specific case where $p = l = 2$, a realization of equation 5.4 is written:

$$\mathbf{x}_t = c_0 + C_1\mathbf{x}_{t-1} + C_2\mathbf{x}_{t-2} + \mathbf{u}_t \tag{5.7}$$

which is expressed as:

$$\begin{bmatrix} x_t^{(1)} \\ x_t^{(2)} \end{bmatrix} = \begin{bmatrix} c_{01} \\ c_{02} \end{bmatrix} + \begin{bmatrix} c_{11}^{(1)} & c_{12}^{(1)} \\ c_{21}^{(1)} & c_{22}^{(1)} \end{bmatrix} \begin{bmatrix} x_{t-1}^{(1)} \\ x_{t-1}^{(2)} \end{bmatrix} + \begin{bmatrix} c_{11}^{(2)} & c_{12}^{(2)} \\ c_{21}^{(2)} & c_{22}^{(2)} \end{bmatrix} \begin{bmatrix} x_{t-2}^{(1)} \\ x_{t-2}^{(2)} \end{bmatrix} + \begin{bmatrix} \epsilon_{1t} \\ \epsilon_{2t} \end{bmatrix}.$$

This can be expressed as a realization of a VAR(1), as shown in equation 5.5, of the form :

$$\underline{\mathbf{X}}_{t:t-1} = \mathcal{C}\underline{\mathbf{X}}_{t-1:t-2} + \underline{\mathbf{U}}_{t:t-1} \tag{5.8}$$

where the realization is written:

$$\underbrace{\begin{bmatrix} 1 \\ \mathbf{x}_t \\ \mathbf{x}_{t-1} \end{bmatrix}}_{\underline{\mathbf{x}}_{t:t-1}} = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ c_0 & C_1 & C_2 \\ 0 & I & 0 \end{bmatrix}}_{\mathcal{C}} \underbrace{\begin{bmatrix} 1 \\ \mathbf{x}_{t-1} \\ \mathbf{x}_{t-2} \end{bmatrix}}_{\underline{\mathbf{x}}_{t-1:t-2}} + \underbrace{\begin{bmatrix} 0 \\ \mathbf{u}_t \\ 0 \end{bmatrix}}_{\underline{\mathbf{u}}_{t:t-1}}. \tag{5.9}$$

and can be further developed as:

$$\begin{bmatrix} 1 \\ x_t^{(1)} \\ x_t^{(2)} \\ x_{t-1}^{(1)} \\ x_{t-1}^{(2)} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ c_{01} & a_{11}^{(1)} & c_{12}^{(1)} & c_{11}^{(2)} & c_{12}^{(2)} \\ c_{02} & a_{21}^{(1)} & c_{22}^{(1)} & c_{21}^{(2)} & c_{22}^{(2)} \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ x_{t-1}^{(1)} \\ x_{t-1}^{(2)} \\ x_{t-2}^{(1)} \\ x_{t-2}^{(2)} \end{bmatrix} + \begin{bmatrix} 0 \\ \epsilon_{1t} \\ \epsilon_{2t} \\ 0 \\ 0 \end{bmatrix}. \tag{5.10}$$

5.2.2.3 NonLinear VAR (NVAR)

A more general formulation of the VAR process is given by the NonLinear VAR as follows:

$$\mathbf{X}_t = F(\underline{\mathbf{X}}_{t-1:t-l}) + \mathbf{U}_t \tag{5.11}$$

where the function $F : \mathbb{R}^{(pl+1)} \to \mathbb{R}^p$ represents an NVAR model and can be rewritten as:

$$\underline{\mathbf{X}}_{t:t-l+1} = \mathbf{F}(\underline{\mathbf{X}}_{t-1:t-l}) + \underline{\mathbf{U}}_{t:t-l+1} \tag{5.12}$$

where:

$$\mathbf{F} = \begin{bmatrix} e_1^\intercal \\ F \\ S_{p(l-1)} \end{bmatrix} = \begin{bmatrix} [1, 0, \ldots, 0] \\ F \\ \begin{bmatrix} 0 & I_{p(l-1)} \end{bmatrix} \end{bmatrix} \tag{5.13}$$

with $e_1 = [1, 0, \ldots, 0] \in \mathbb{R}^{pl+1}$ and $S_{p(l-1)}$ is the upper shift matrix of size $\mathbb{R}^{p(l-1)\times(pl+1)}$, and the 1's are located on the first upper diagonal.

This formulation allows for non-linear relationships between the variables (Morioka et al., 2021).

Remark 5.2.2. *The VAR process is a specific type of NVAR process where* $F(\underline{\mathbf{X}}_{t-1:t-l}) = \mathcal{C}\underline{\mathbf{X}}_{t-1:t-l}$.

VAR models are parameterized by two key parameters: the number of variables p to include in the system and the lag order l. Their choice influences both computational complexity and forecasting errors due to the number of estimated coefficients (Hyndman & Athanasopoulos, 2018).
These models have the ability to capture complex inter-dependencies between variables with flexible modeling of the relationships without requiring prior assumptions on the underlying time series structure. Additionally, VAR formulation allows for simultaneous estimation of all variables' relationships and facilitates Granger causality testing.
However, challenges arise in high-dimensional data sets where estimation complexity increases significantly. In addition, the large number of relationships captured by their coefficients can hinder interpretability, limiting our ability to understand the underlying relationships. For a more detailed discussion on the evolution of VAR models, their strengths, and limitations, please refer to (De Gooijer & Hyndman, 2006).

The next part explores different strategies employed in time series forecasting to predict future observations across multiple future time steps. This extends the scope of forecasting beyond the immediate next value in a time series.

5.2.3 Time Series Forecasting Strategies

Learning an effective forecasting model from temporal data plays an important role in a variety of fields, as it enables the prediction of future trends and aids decision-making. The structure of this model is determined by the specific objective and time horizon chosen. In the following, we describe two primary strategies used in the literature for both univariate and multivariate time series forecasting (Ben Taieb et al., 2012; De Stefani, 2022).

The objective is to learn a function $\hat{\mathbf{F}}$ that predicts the future $(\mathbf{x}_{t+H:t+1})$ given the past $(\mathbf{x}_{t:t-l+1})$.

5.2.3.1 Single-step estimation

In single-step forecasting, the goal is to estimate a function $\hat{\mathbf{F}} : \mathbb{R}^{pl+1} \to \mathbb{R}^{pl+1}$ that predicts the next observation at time $t+1$ of the time series, given the past l observation up to the current time t, where l is an estimate of the true lag order l. The prediction at time $t+1$, denoted as $\underline{\hat{\mathbf{x}}}_{t+1:t-l+2} \in \mathbb{R}^{pl+1}$, represents the predicted future values given information up to time t, i.e $\underline{\mathbf{x}}_{t:t+1-l}$. The single-step prediction is formulated as follows:

$$\underline{\hat{\mathbf{x}}}_{t+1:t-l+2} = \hat{\mathbf{F}}(\underline{\mathbf{x}}_{t:t+1-l}) \tag{5.14}$$

In this setting, the risk is expressed as :

$$\mathbb{E}_{\mathcal{D}}\left[\left\|\underline{\mathbf{x}}_{l+1:2} - \hat{\mathbf{F}}(\underline{\mathbf{x}}_{l:1})\right\|^2\right]$$

5.2.3.2 Multi-step estimation

Multi-step forecasting aims to predict multiple future values, going beyond a single step. It seeks to forecast future values up to time $t+H$, where $H \in \mathbb{N}$ is the forecast horizon, based on l observations up to the current time step t. This problem is obviously more difficult than the one-step problem. In fact, predicting the long-term horizon increases uncertainty. In the following, we present the main strategies (Ben Taieb et al., 2012) that have been developed for this objective.

- **Recursive approach** (or iterative approach) tackles multi-step forecasting by estimating a single one-step forecasting model $\hat{\mathbf{F}} : \mathbb{R}^{pl+1} \to \mathbb{R}^{pl+1}$ and by iteratively performing one-step predictions for each future time step, up to a specified horizon H. At each step, the previously predicted value is fed back into the function as input to predict the next value in the sequence. Using time series data, the recursive multi-step prediction can be written as follows:

  $$\underline{\hat{\mathbf{x}}}_{t+h:t+h-d+1} = \hat{\mathbf{F}}^h(\underline{\mathbf{x}}_{t:t-l+1})$$

  where $\hat{\mathbf{F}}^h$ is the composition of the function $\hat{\mathbf{F}}$.

  The risk can be expressed as :

  $$\sum_{h=1}^{H} \mathbb{E}_{\mathcal{D}}\left[\left\|\underline{\mathbf{x}}_{l+h:h+1} - \hat{\mathbf{F}}^h(\underline{\mathbf{x}}_{l:1})\right\|^2\right]$$

  The advantage of this approach is that it ensures that the fitted model is close to the true function $\mathbf{F}$ generating the data if it exists. However, there is no necessary guarantee of the minimization of the multi-step forecast error (discrepancy between the true future values and the multi-step forecasts). This is partly due to errors introduced at each prediction step, which can accumulate and propagate over longer horizons.

- **Direct approach** tackles multi-step forecasting by independently estimating different models $\hat{\mathbf{F}}_h : \mathbb{R}^{pl+1} \to \mathbb{R}^{pl+1}$ for each forecast horizon $t+h$, for $h \in \{1, \ldots, H\}$. Each model directly predicts the future value at horizon h, using only past observations. Using time series data, the direct multi-step prediction can be written for each forecast horizon $h \in \{1, \ldots, H\}$, as follows:

$$\underline{\hat{\mathbf{x}}}_{t+h:t+h-l+1} = \hat{\mathbf{F}}_h(\underline{\mathbf{x}}_{t:t-l+1}) \tag{5.15}$$

  In this setting, the risk is expressed as :

$$\sum_{h=1}^{H} \mathbb{E}_{\mathcal{D}} \left[\left\| \underline{\mathbf{x}}_{l+h:h+1} - \hat{\mathbf{F}}_h(\underline{\mathbf{x}}_{l:1}) \right\|^2 \right]$$

  This strategy suggests that individual models might require different parameter sets to capture the specific relationships for each horizon. Additionally, the optimal lag order for each model might differ. Instead of aiming to match the potential underlying DGP $\mathbf{F}$, the direct strategy prioritizes achieving accurate forecasts using separate models for each horizon. This comes at the cost of increased computational complexity due to the need to train and evaluate multiple models (Ben Taieb, 2014).

Remark 5.2.3. *A more general approach is to express risk as a weighted sum over the forecast horizon as:*

$$\sum_{h=1}^{H} \mathbb{E}_{\mathcal{D}} \left[w_h \left\| \underline{\mathbf{x}}_{l+h:h+1} - \hat{\mathbf{F}}_h(\underline{\mathbf{x}}_{l:1}) \right\|^2 \right]$$

Indeed, a smaller weight could be attributed to a larger horizon as the uncertainty increases.

Additional approaches have been developed (Bontempi et al., 2013), including the MIMO (multi-input multi-output) approach (Bontempi, 2008), which forecasts all future steps (i.e., H steps) simultaneously.

Following Ben Taieb's work (Ben Taieb et al., 2012; Petropoulos et al., 2022), there is no general rule about whether the direct or recursive approach is better, as this choice is a tradeoff between forecast bias and variance. Consequently, the question of the optimal approach remains open and often requires empirical evaluation to determine the best method for a specific situation.

5.3 From forecasting to dynamic discovery

While accuracy remains the primary objective in forecasting, complementary objectives such as understanding the underlying mechanisms and modeling the relationships between features within the time series have also gained attention. In dynamic modeling, and particularly in discrete-time methods, the goal is to characterize the evolution of a system over discrete time intervals. One approach is to leverage interpretable models to uncover the underlying dynamics and gain insights into the relationships between the involved variables while performing accurate forecasts.

5.3.1 Problem Definition

Building on the concepts introduced in Section 2.4.2.4, this section formalizes the dynamic modeling problem as follows. The objective is to identify an interpretable forecasting model, denoted by $\hat{\mathbf{F}}$, from a class of functions $\mathcal{F}$ that minimizes the empirical error (e.g., mean squared error) using the available time series data $\mathbf{x}_{T:1}$. The class $\mathcal{F}$ may encompass various statistical and ML models for time series forecasting, such as VAR, ARIMA, and exponential smoothing.

In this setting, we aim to find a function, denoted as $\hat{\mathbf{F}}$, that solves the following minimization problem:

$$\hat{\mathbf{F}} = \underset{\tilde{\mathbf{F}} \in \mathcal{F}}{\operatorname{argmin}} \; \mathbb{E}_{\mathcal{D}} \left[\left\| \underline{\mathbf{x}}_{l+1:2} - \tilde{\mathbf{F}}(\underline{\mathbf{x}}_{l:1}) \right\|^2 \right]$$

where $\mathcal{F}$ is a class of functions mapping from $\mathbb{R}^{pl+1}$ to $\mathbb{R}^{pl+1}$. The most straightforward approach is to minimize its empirical counterpart

$$\hat{\mathbf{F}} = \underset{\tilde{\mathbf{F}} \in \mathcal{F}}{\operatorname{argmin}} \left\{ \sum_{t=d}^{T-1} \sum_{i=1}^{N} \| \underline{\mathbf{x}}_{i,t+1:t-l+2} - \tilde{\mathbf{F}}(\underline{\mathbf{x}}_{i,t:t-l+1}) \|^2 \right\}. \tag{5.16}$$

It is, however, well known that this approach may suffer from overfitting: the minimizer may perform well on the observed samples but fail to generalize to the underlying dynamic we try to model. To cope with this issue, some regularization is required. The most classical formulation is the regularized minimization problem

$$\hat{\mathbf{F}} = \underset{\tilde{\mathbf{F}} \in \mathcal{F}}{\operatorname{argmin}} \left\{ \sum_{t=d}^{T-1} \sum_{i=1}^{N} \| \underline{\mathbf{x}}_{i,t+1:t-l+2} - \tilde{\mathbf{F}}(\underline{\mathbf{x}}_{i,t:t-l+1}) \|^2 + \lambda C(\tilde{\mathbf{F}}) \right\}, \tag{5.17}$$

where $C(\tilde{\mathbf{F}})$ is a measure of complexity of $\tilde{\mathbf{F}}$ and λ is a parameter to be tuned that balances a trade-off between minimizing the loss and controlling the complexity of the function to avoid overfitting. Other formulations are possible: one may explicitly restrict $\tilde{\mathbf{F}}$ to have complexity $\mathcal{C}(\tilde{\mathbf{F}})$ smaller than a threshold η or constrain this complexity through algorithmic design.

5.3.2 Challenges

Estimating the underlying dynamic presents several challenges. Firstly, the lack of prior knowledge necessitates choosing a flexible function class capable of handling both linear and non-linear relationships within the data. However, this flexibility also carries the risk of model misspecification, where the chosen model does not accurately capture the true governing function $\mathbf{F}$ if it exists. Secondly, the estimation method must deal with high-dimensional multivariate time series data, increasing the complexity and the effect of multicollinearity where multiple features are highly correlated. Although multicollinearity may not affect the model's predictive ability, it can hinder its capacity to identify the correct coefficients of interest (Shmueli, 2010). Additionally, incorporating lagged variables becomes essential for large temporal dimensions ($T \gg 1$). However, selecting the optimal number of lagged variables plays a key role in achieving this effectively, as inappropriately chosen lags can lead to sub-optimal performance. Finally, the methods should be computationally efficient as a significant amount of data is often required for accurate estimation.

In the following section, we present a state-of-the-art overview of XAI forecasting approaches, with a focus on interpretable models.

5.4 State of the art in dynamic modeling

In this section, we analyze function classes $\mathcal{F}$ that can be categorized into sparse regression-based or symbolic regression-based approaches and describe the methods in the single-step forecasts setting.

5.4.1 Sparse Regression

Dynamic modeling generally assumes that a small number of relevant variables, taken from a high-dimensional space, govern the underlying system (e.g., physical systems). Techniques have been built upon this idea, leveraging sparse regression and compressed sensing techniques to extract simplified equations that capture the system's underlying dynamics (Brunton et al., 2016; Donoho, 2006). In the following, we detail the preprocessing steps required for this analysis, followed by an exploration of the main approaches for learning these sparse equations from data.

5.4.1.1 Data Preprocessing

Library of function

While linear models offer a basic framework for time series forecasting, they often fail to capture the data's complex relationships and non-linear dynamics. To address these limitations, a popular approach is incorporating various non-linear features into the model. Let $\Phi : \mathbb{R}^{pl+1} \to \mathbb{R}^{p_\Phi}$ denote a function that transforms the original input features of length p into a new set of p_Φ features, where $p_\Phi > pl + 1$. Hence, the input features $\underline{\mathbf{x}}_{t:t+1-l}$ are transformed as follows:

$$\Phi(\underline{\mathbf{x}}_{t:t-l+1}) = \begin{bmatrix} 1 & \Phi_1(\mathbf{x}_{t:t-l+1}) & \Phi_2(\mathbf{x}_{t:t-l+1}) & \dots & \Phi_{p_\Phi}(\mathbf{x}_{t:t-l+1}) \end{bmatrix}^\intercal \tag{5.18}$$

The types of libraries include polynomial functions like powers and cross-products, as well as trigonometric functions (e.g., cosine, sine), exponential functions, and even frequency-domain representations like the Fourier transform.

Example 5.4.1. For $p = d = 2$ with $\underline{\mathbf{x}}_{t:t-1} = [1, x_t^{(1)}, x_t^{(2)}, x_{t-1}^{(1)}, x_{t-1}^{(2)}]$, the polynomial expansion up to degree 2 is written:

$$\begin{aligned}\Phi(\underline{\mathbf{x}}_{t:t-1}) = \Big[&1, x_t^{(1)}, x_t^{(2)}, x_{t-1}^{(1)}, x_{t-1}^{(2)}, \left(x_t^{(1)}\right)^2, \left(x_t^{(2)}\right)^2, \left(x_{t-1}^{(1)}\right)^2, \left(x_{t-1}^{(2)}\right)^2, \\ &x_t^{(1)}x_t^{(2)}, x_{t-1}^{(1)}x_{t-1}^{(2)}, x_{t-1}^{(1)}x_t^{(1)}, x_{t-1}^{(2)}x_t^{(2)}, x_{t-1}^{(1)}x_t^{(2)}, x_t^{(1)}x_{t-1}^{(2)}\Big]^\intercal\end{aligned} \tag{5.19}$$

Features scaling

Depending on the choice of the model, scaling the input features may be a necessary preprocessing step before fitting. Methods such as penalized regression require scaling, as the objective function depends on the scale of the variables. This step ensures that each variable is treated equivalently.

Various types of scaling can be applied to the input features. One common approach is normalization, which centers each column by subtracting its mean and then dividing by its standard deviation, ensuring that each variable has a mean of zero and a standard deviation of

one. Another scaling approach is dividing each feature vector by its l_2 norm, ensuring that all features contribute equally, as each vector has a norm of one. In the following, we assume that the features are scaled.

5.4.1.2 Feature selection

Dynamic modeling typically relies on the assumption of a limited number of informative variables. Therefore, when working with high-dimensional data with multicollinearity, especially after incorporating non-linear features using a function library Φ, the objective is to select a sparse subset of relevant variables. The techniques used should favor estimators that are consistent [1]. There are two types of consistency (Bach, 2008):

- **Parameter consistency**: This ensures that the estimated coefficient values are close to the true ones as the data size increases.
- **Model Consistency**: This ensures that non-zero true coefficients are estimated as non-zeros.

To address this challenge, feature selection methods aim to select a subset of input features that maximize the performance of the model (Bontempi, 2021). There are three main approaches:

- **Filter methods** rank features according to a metric or relevance score S, which is a real-valued function, and then select the top $k \in \{1, \dots, p\}$ features. It is expressed as:

$$\underset{|s|=k, s\subset\{1,\dots,p\}}{\operatorname{argmax}} \sum_{t=d}^{T-1} S(\underline{\mathbf{x}}^{(s)}_{t:t-l+1}, \underline{\mathbf{x}}_{t+1:t-l+2}). \tag{5.20}$$

  where $\mathbf{x}^{(s)}_{t:t-l+1} \in \mathbb{R}^{sl+1}$ is the vector containing the features of $\mathbf{x}_{t:t-l+1}$ indexed by elements in the subset s. The relevance score can be based on different statistical measures, such as mutual information or cross-correlation. Alternatively, some filter methods select features based on a percentile of the highest scores, avoiding the need to pre-specify the exact number of features k to be selected.

- **Wrapper methods** use a search algorithm to find the subset of features that maximizes the model's performance. The optimization problem can be expressed as:

$$\underset{|s|\leq k}{\operatorname{argmin}} \sum_{t=l}^{T-1} \sum_{i=1}^{N} \|\underline{\mathbf{x}}_{i,t+1:t-l+2} - \tilde{\mathbf{F}}(\underline{\mathbf{x}}^{(s)}_{i,t:t-l+1})\|^2,$$

  where $\|.\|$ represents the objective function, which quantifies the model's performance using various metrics such as Root Mean Squared Error (RMSE). Search methods employed to find the optimal solution that minimizes this quantity can be broadly categorized into three types: exhaustive, greedy, and randomized.

- **Embedded Methods** integrate feature selection directly into the model's learning process. In the next part, we explore specific examples of embedded methods, including regularized methods.

[1] A statistical estimator is consistent if $\hat{\mathbf{F}} \to \mathbf{F}$ as $T \to \infty$ converge to the true function/value as the data size increases.

Sparse regression is a popular technique in dynamic modeling that reduces model complexity by selecting only the most influential variables. The representation of time series data using the VAR framework allows using the penalized methods, described in section 2.4.2.1, to achieve sparsity. This section dives deeper by exploring the implementation details of the penalization techniques, discussing alternative approaches, and analyzing their practical applications and limitations.

5.4.1.3 Penalized Regression

Penalized regression techniques address variable selection and model fitting simultaneously. These methods achieve this by incorporating a penalty term into the objective function, which penalizes the magnitude of the estimated regression coefficients. The problem can be defined as:

$$\hat{\mathcal{C}} = \underset{\mathcal{C}}{\operatorname{argmin}} \quad \left\{ \frac{1}{T-l+1} \sum_{t=d}^{T} \|\underline{\mathbf{x}}_{t:t-l+1} - \mathcal{C}\underline{\mathbf{x}}_{t-1:t-l}\|_2^2 + \lambda R(\mathcal{C}) \right\}, \tag{5.21}$$

where $\|\cdot\|_2$ is the ℓ_2 norm, $\mathcal{C} \in \mathbb{R}^{(pl+1)\times(pl+1)}$ is a coefficient matrix as defined in equation 5.5, R is a regularization function and λ the regularization parameter. By tuning the penalty parameter, we can control the trade-off between model complexity (e.g., number of variables) and goodness-of-fit.

The objective is to reduce the complexity and **simplify** the model represented by the coefficient matrix $\mathcal{C}$. Since each coefficient in $\mathcal{C}$ accounts for the influence of a variable in $\underline{\mathbf{x}}_{t-1:t-l}$ on the target, reducing the number of non-zero elements in $\mathcal{C}$, making the matrix sparse, effectively reduces the number of parameters considered by the model.

l_0 penalty

The main approach to achieve sparsity is formulated using the l_0 penalty with $R(\mathcal{C}) = \|\mathcal{C}\|_0$, which accounts for the number of non-zero components. As seen in section 2.4.2.1, due to the discontinuity and non-convexity of the penalty, it is a combinatorial problem that is NP-hard. Consequently, there is no efficient algorithm that guarantees the finding of optimal solutions, especially in high dimensions. However, several approaches have been developed to find approximate solutions (Huang et al., 2018).

l_1 penalty

One common approach is to relax this non-convex problem to a convex one by replacing the penalty with an l_p norm where $p \in (0, 1]$. In this context, the most popular is the Lasso regression using l_1 norm (Tibshirani, 1996) where $R(\mathcal{C}) = \|\mathcal{C}\|_1$. Several algorithms have been developed to solve this problem, including Least Angle Regression (LARS) (Osborne et al., 2000), coordinate descent (Wu & Lange, 2008) and proximal gradient descent(Agarwal et al., 2010).

However, the Lasso is not a consistent estimator, implying that it does not converge to the true coefficient of the underlying system. Additionally, multicollinearity can further challenge its ability to produce consistent estimates. Several authors have established conditions under which the Lasso can identify the true underlying system (Hastie et al., 2009) (p. 91) (model consistency) and (Zhao & Yu, 2006) stated the *irrepresentable condition* which provides a

necessary and sufficient condition on the design matrix for the Lasso to exhibit model consistency. Multicollinearity remains a big difficulty for lasso to be consistent. We invite the reader to read (Hastie et al., 2009; Huang et al., 2018) for further details.

Heuristic

Another common approach is to take advantage of a greedy algorithm that seeks sequentially for local optimum with the objective of reaching a global optimum. There are two main methods described in the literature for achieving this goal. Both methods begin by estimating a matrix $\hat{\mathcal{C}}$, which is the solution to equation 5.21. Then, each approach is defined, and its steps are outlined as follows:

- **Thresholding approach**: This technique involves setting a threshold to the learned coefficient to induce sparsity. After estimating $\hat{\mathcal{C}}$, the algorithm applies a threshold to enforce sparsity. Using notation in equation 5.6, coefficients are represented in the matrices C_k from the estimated matrix $\hat{\mathcal{C}}$ with $k \in \{1, \dots, d\}$. For each estimated coefficient c in C_k, the **hard thresholding** is written:

$$c^* = \begin{cases} 0, & \text{if } c < \eta \\ c, & \text{otherwise} \end{cases} \tag{5.22}$$

  where $\eta \in \mathbb{R}$ denotes a threshold. It can be either pre-defined as a hyperparameter or learned during the training process along with other model parameters. The remaining non-zero coefficients of the matrix $\hat{\mathcal{C}}$ after thresholding define a new matrix $\hat{\mathcal{C}}^*$, which allows selecting the relevant variables in the model.

- **Iterative/Sequential Learning approach**: This technique iteratively refines a sparse model. At each step, it focuses only on the coefficients identified as non-zero by the previous sparse model and refits them. This process iteratively increases the sparsity of the model, simplifying it by relying on smaller subsets of coefficients.

  We define the support of $\hat{\mathcal{C}}$, denoted as $\text{supp}(\hat{\mathcal{C}})$, as the set of indices of non-zeros elements of $\hat{\mathcal{C}}$: $\text{supp}(\hat{\mathcal{C}}) = \{(j,k) \in \{1, \dots, pl+1\}^2 | \hat{\mathcal{C}}_{j,k} \neq 0\}$, where $\hat{\mathcal{C}}_{j,k}$ is the element of $\hat{\mathcal{C}}$ in the j-th line and k-th column. Let $\mathcal{M}(\hat{\mathcal{C}})$ be the space of matrices with the same sparsity pattern as $\hat{\mathcal{C}}$, i.e., the set of all matrices that have the same support as $\hat{\mathcal{C}}$. It is defined as:

$$\mathcal{M}(\hat{\mathcal{C}}) = \{M \in \mathbb{R}^{(pl+1)\times(pl+1)} | \text{ supp}(M) = \text{supp}(\hat{\mathcal{C}})\} \tag{5.23}$$

  We then fit a simplified model on the space $\mathcal{M}(\hat{\mathcal{C}})$:

$$\hat{\mathcal{C}}^* = \underset{C \in \mathcal{M}(\hat{\mathcal{C}})}{\text{argmin}} \left\{ \frac{1}{T-l+1} \sum_{t=l}^{T} \|\underline{\mathbf{x}}_{t:t-l+1} - C\underline{\mathbf{x}}_{t-1:t-l}\|_2^2 + \lambda R(\mathcal{C}) \right\} \tag{5.24}$$

Several algorithms leverage these approaches to address limitations in Lasso regression, particularly the bias in coefficient estimation (Hastie et al., 2009)(p. 91). These techniques include:

- Two-step approaches that first perform variable selection in the first step using Lasso. Subsequently, the selected variables are used in a second step to fit a model, typically Ordinary Least Squares (OLS), for obtaining unbiased coefficient estimates (e.g., (Wipf & Nagarajan, 2010)).
- Refined Lasso such as the relaxed Lasso (Meinshausen, 2007). It first uses Lasso for initial variable selection, followed by a reapplication of Lasso on the selected subset of variables. This allows for increased sparsity and potentially increased performance.

Several other methods are also widely used. Among them, the Forward (backward) step-wise selection is an algorithm that starts with the intercept (with the full model) and sequentially adds (removes) to the model variables that improve the fit (Hastie et al., 2009). Additionally, Iterative Hard Thresholding (Blumensath & Davies, 2009) is an iterative method that alternates between gradient descent steps and hard thresholding steps that set all estimated values below a threshold to zero.

Sequential thresholded Least-squares algorithm Among sparse regression and state-of-the-art methods for dynamic discovery, SINDy (Sparse Identification of Nonlinear Dynamics) developed by Brunton et al. (2016) stands out for its ability to capitalize on the previously described approaches. This technique facilitates the identification of the equations governing complex systems. By assuming sparsity in nonlinear differential equations, Brunton et al. (2016) take advantage of sparse regression techniques to identify the equations from data. To overcome the limitation of Lasso, which is computationally expensive for very large data sets, the Threshold Least Square algorithm was developed, which is also a relaxation of the l_0 minimization problem. It involves applying recursively two steps:

1. **Least square**: The first step of the algorithm estimate the coefficient matrix $\hat{\mathcal{C}}$ that solves the least square problem.
2. **Thresholding**: In the second step, the algorithm applies a threshold to enforce sparsity. The remaining non-zero coefficients of the matrix $\hat{\mathcal{C}}$ allow selecting the variables to keep for the next iterations.

The recursion continues until convergence, i.e., the identification of the non-zero coefficients. The threshold is either a parameter of the model to learn or a hyperparameter to estimate by cross-validation, for example.

Note that the sequential thresholded least squares algorithm can be viewed as a particular instance of the sparse relaxed regularized regression (SR3) problem (Champion et al., 2020; Zheng et al., 2018).

5.4.1.4 Related approaches

Imitation Learning In imitation learning, the objective is to train an agent through expert demonstration. The objective is for the agent to learn the underlying policy, which represents the decision-making process used by the expert. By analyzing the demonstrations, which consist of a data set, the agent learns to **imitate** the expert's behavior. Several algorithms have been proposed for imitation learning, including DAgger (data-set Aggregation) and

SMILe (Stochastic Mixing Iterative Learning) (Ross & Bagnell, 2010; Ross et al., 2011)). The field of imitation learning can be a valuable tool for time series dynamic discovery. Indeed, Venkatraman et al. (2015) pioneered this approach in the context of multi-step prediction, with the algorithm *Data As Demonstrator* that builds on DAgger. In this framework, the training data is seen as the expert demonstrating the system's dynamics, which the algorithm tries to learn. Venkatraman et al. (2014) demonstrate this approach in practice and also combine it with Subspace Identification (Van Overschee & De Moor, 2012) to extract the dynamics of several systems.

Besides, probabilistic machine learning offers alternative approaches, especially Gaussian processes, to discover equations from data (Duncker et al., 2019; Raissi et al., 2017).

5.4.2 Discrete Search space: Symbolic Regression

Symbolic regression [2] is a powerful technique used to search in a discrete space of mathematical expressions to find an equation that best fits the data set.

5.4.2.1 Representation

Symbolic regression usually represents the discovered mathematical equation as a binary expression tree or syntax tree. It is a type of data structure specifically designed for this objective. The nodes in the tree are of two types:

- **Internal nodes** represent algebraic **operators** such as addition, multiplication, division and substraction $(+, \times, -, /)$ and analytical **function** like exponential or cosine functions.
- **Leaf nodes** represent the operands, i.e., the variable to use in the equation. These nodes do not have children

The tree structure is formed by edges that connect internal nodes to their descendants (including internal and leaf nodes), establishing the order of operations. An example of an expression tree is presented in Figure 5.1.

5.4.2.2 Evolutionary Algorithms

Evolutionary Algorithms are a class of search-based optimization algorithms. Among them, genetic programming (Ferreira, 2006; Koza, 1994) explores a search space of equations and selects the most suitable given the data without prior structure knowledge. Indeed, from a library of operators and functions, the genetic or evolutionary algorithms mimic the process of natural selection to evolve individuals, i.e., equations/trees, through successive generations. The algorithm **initializes** with a population of randomly generated equations and **evaluates** their data fit based on a fitness function. In the context of dynamic discovery, the fitness function can be based on how well the equations capture the underlying dynamic with a loss such as the RMSE. Then, genetic algorithms typically iterate until convergence over several steps, such as:

[2] There are broader definitions of "symbolic regression" that encompass any equation discovery. Here, we restrict our definition to methods that explore a discrete space of mathematical expression to retrieve equations.

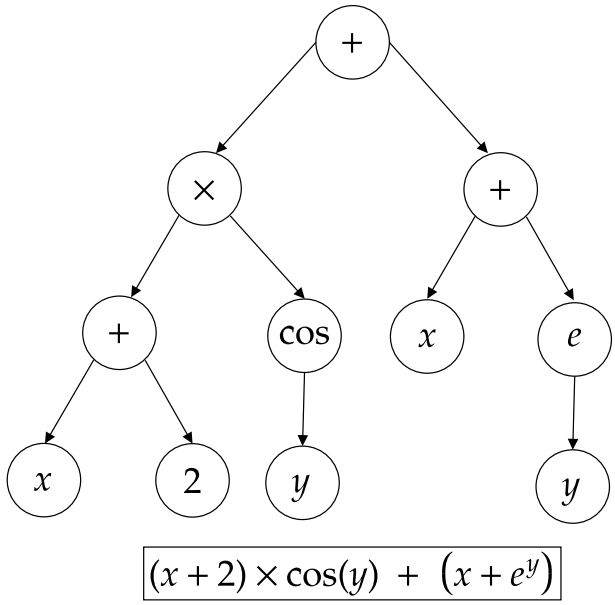


Figure 5.1: Symbolic regression model in mathematical notation and in expression tree representation

- **Selection**: Select the individuals, i.e., equations from the population, to serve as parents for the population of the next generation (i.e., iteration). This is typically done by selecting individuals with the highest fitness score.
- **Crossover**: This step involves combining two or more individuals to create new "offspring". Indeed, it creates new equations by combining the parts of two or more equations from the populations.
- **Mutation**: Mutation introduces random changes in selected individuals from the population in order to promote diversity and prevent premature convergence. Typically, components of the selected equations are modified and generate a new individual for the next generation.

These steps are repeated until a stopping criterion is met. The resulting equation represents the discovered relationships between variables from the data.

Genetic programming is the most popular algorithm used in symbolic regression with pioneering work of Schmidt & Lipson (2009) with the aim to discover physical equations from data. A large body of work have been explored to this day in this domain, for a more comprehensive study refer to (La Cava et al., 2021; Makke & Chawla, 2024). The limitation of GP-based methods is that they do not scale in high dimensional settings and are highly sensitive to hyperparameters (Makke & Chawla, 2024; Petersen, 2019).

5.4.2.3 Other approaches

Researchers are also tackling symbolic regression with other approaches. This includes deep learning methods such as the AI-Feynman algorithm (Udrescu & Tegmark, 2020), which

leverages physics-inspired concepts like symmetry and dimensional analysis, reinforcement learning techniques such as Deep Symbolic Regression (Petersen, 2019), game playing techniques such as Monte-Carlo tree search (Sun et al., 2023) and Bayesian methods (Guimerà et al., 2020). More details and additional approaches can be found in (Camps-Valls et al., 2023; Makke & Chawla, 2024).

5.4.3 Conclusion

The field of dynamic discovery is active and presents many challenges on both theoretical and practical levels. On the theoretical level, the identifiability of the underlying dynamics and the adoption of modeling criteria such as parsimony or the choice of evaluation criteria are open problems (Camps-Valls et al., 2023). Moreover, studies often rely on two strong assumptions that are not always verified: causal sufficiency, i.e., the presence of all variables in the dataset, and representativeness, i.e., that the dataset represents the true dynamics. On the practical side, high dimensionality, multicollinearity, system non-linearity, and the risk of overfitting are all problems encountered in dynamic modeling.

5.5 How our work fits in the literature

Following identified challenges in the field of dynamic modeling (Camps-Valls et al., 2023), we leverage interpretable models with the dual advantage of revealing the underlying dynamics of the system, as well as having predictive capabilities. Our work focuses on improving sparse regression methods like SINDy (Brunton et al., 2016) by addressing errors that arise during the training process. While sharing a common goal with Taieb & Hyndman (2012)'s work of correcting forecasting errors, our approach focuses on building a single forecasting model through a recursive approach to uncover the underlying dynamic of the system. In the following chapter, we propose an interpretable and coherent multi-step forecasting method for multivariate time series by composing a single one-step linear predictor as an NVAR model. Inspired by the Data as Demonstrator (Venkatraman et al., 2015), an extension of DAgger (Ross & Bagnell, 2010), we propose a solution using an original iterative algorithm involving simple least squares problems and augmented training data sets.

CHAPTER 6

Interpretable Forecasting Model

Contents

6.1 Introduction

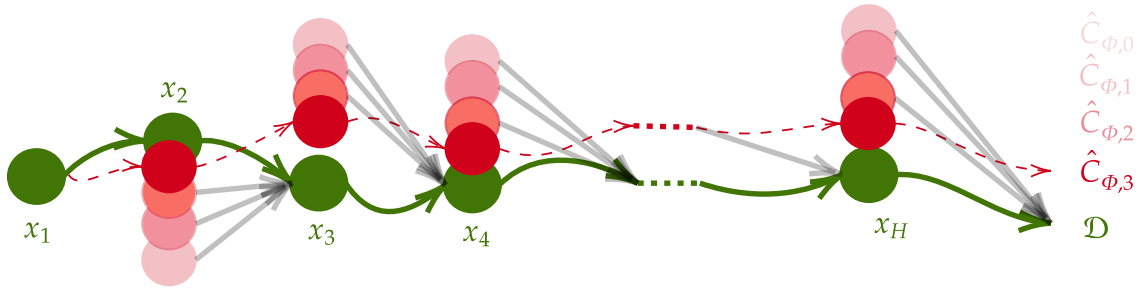
Many real-life phenomena that one seeks to predict are governed by equations involving several variables and depend on time trends. For example, the weather depends on temperature, atmospheric humidity, and wind, which in turn depend on variations in atmospheric pressure, altitude, and relief. In these systems, the prediction must be not only accurate but also simultaneous and coherent between all variables. From multivariate time series, one of today's challenges is to determine the underlying dynamics, which may, for example, be a physical equation governing evolutions. Thus, the objective is to learn the invariant properties and characteristics from the data.

Furthermore, in domains such as climate (Bouche et al., 2022; Liu et al., 2019), energy (Hadri et al., 2021), healthcare (Lim, 2018) or finance (Capistrán et al., 2010), it is necessary to build a model with the ability to predict in the short, medium and long term. Hence, we want a model that can make multi-step predictions while learning the system's dynamics. To perform multi-step ahead prediction, there are two main strategies: first, the direct approach builds for each prediction horizon an independent model, whereas the second, the recursive approach, learns a one-step predictor and iterates using the previous predictions as input. The direct method has limitations since it cannot learn the statistical dependencies between the different prediction horizons. Hence, the recursive method seems more natural as it predicts from one instant to another by composing the same model keeping a *coherence* and the existing statistical dependencies from one instant to another. Consequently, in the example of a physical system, the objective would be to obtain a predictor by learning the parameters of the equation of motion from the data.

Traditionally, time series modeling uses linear models from statistics or econometrics, such as autoregressive models (Hamilton, 1994; Sims, 1980). These models have been widely used because they are practical and have mathematical guarantees. However, these models suffer from difficulties in predicting the long term in the case of nonlinear data. Indeed, as soon as a prediction error is made, given that this approach is recursive, the model will take this error as input in order to predict the next time generating an error propagation. Many machine learning models have been developed to overcome these limitations, such as Recurrent Neural Networks (Dabrowski et al., 2020; Lim et al., 2021; Salinas et al., 2020). Despite their prediction capacity, they have a major drawback: the lack of interpretability. Although there are meta-models of decision explanations (Lundberg & Lee, 2017; Ribeiro et al., 2016), they remain insufficient for our objectives. Besides, models such as transformers (Choi et al., 2016; Li et al., 2019; Lim et al., 2021) with properties inherent to their architectures can explain certain characteristics of the time series, such as point changes or trends, but do not allow for finding the system dynamics.

To our knowledge, the closest work to ours is that of Venkatraman et al. (2015) on the Data as Demonstrator (DaD) algorithm. This paper builds on DAgger (Ross et al., 2011), an imitation learning algorithm, by proposing a meta-model to improve multi-step performance. Using a one-step predictor, the algorithm iteratively increases the data set with the help of the prediction errors during the training phase. Our work is, therefore, an extension and a consolidation of this approach. It allows not only to improve the performances of multi-step predictions but also to learn the dynamics of multivariate time series by adding an interpretability constraint. Moreover, we develop the theoretical guarantees in a thorough way.

Figure 6.1: The figure illustrates the algorithm's ability to learn and iteratively correct trajectories. The aim is to uncover the true underlying dynamic represented by the unknown distribution $\mathcal{D}$ based on the observed time series in green. Initially, the algorithm learns a recursive model, $\mathcal{C}_{\Phi,0}$, that propagates errors. In the subsequent iterations, the algorithm uses previous trajectories to augment training data to progressively improve the model's learning of the true dynamic. Here, the figure depicts the learning process up to the third iteration, where the final model is denoted as $\mathcal{C}_{\Phi,3}$.



The main contributions in this chapter are the following.

- We propose an interpretable and coherent multi-step forecasting method for multivariate time series. It is obtained using an iterative algorithm inspired by DAgger, an extension of DaD.
- We support the use of our method with theoretical guarantees obtained by interpreting it as a Follow-The-Leader algorithm, an online learning algorithm that iteratively learn a model by minimizing the cumulative loss from all previous iterations. The name reflects the strategy of "following" the best-performing model from previous iterations.
- We apply our method to discrete dynamical systems, showing an improvement in multi-step forecasting compared to sparse models trained at one step. We get interpretability by learning a sparse model.
- We apply our method to systems of ordinary differential equations (ODEs) observed at discrete times, showing an improvement in short and long-term predictions.

6.2 Multi-step Forecasting

Our study aims to learn a coherent and interpretable multi-step ahead prediction model. From past information, we want to predict the future evolution up to a given horizon at a given time.

6.2.1 Notations

Let N, H, and $p \in \mathbb{N}$ denote respectively the number of time series of the training data set, the prediction horizon, and the dimension (i.e., the number of variables). For each multivariate time series $(\mathbf{x}_t)_{t=1}^{T} \in \mathbb{R}^{p\times T}$, $(\tau_1, \tau_2) \in \{1, \dots, T\}^2$ such that $\tau_1 < \tau_2$, we denote $\mathbf{x}_{\tau_2:\tau_1} = (\mathbf{x}_t)_{t=\tau_1}^{\tau_2}$.

Hence, for $\tau \in \{1, \dots, T\}$, $\mathbf{x}_{\tau-1:1}$ contains the past values of $\mathbf{x}_\tau$ and $\mathbf{x}_{t:t-l+1}$ contains the l lags of $\mathbf{x}_{t+1}$ (for $l \geq 1$).

Consider N multivariate time series $(\mathbf{x}_{i,T:1}) = (x^{(1)}_{i,T:1}, \dots, x^{(p)}_{i,T:1})$, $i \in \{1, \dots, N\}$, generated by the same stationary process. First, let us define $\underline{\mathbf{x}}_{t:t-l+1}$, the vector containing the past l values and a term 1 allowing to encode the intercept as described in equation 5.6. In each time series, we extract pairs of sequences $\mathbf{z}_{i,t} = ((\underline{\mathbf{x}}_{i,t:t-l+1}), (\underline{\mathbf{x}}_{i,t+H:t+1}))$ for $t \in \{l, \dots, T-H\}$. All the $\mathbf{z}_{i,t}$ have the same distribution $\mathcal{D}$ and we denote by $\mathcal{D}_N = \{\mathbf{z}_{i,t} : i = \{1, \dots, N\}, t = \{l, \dots, T-H\}\}$ the set of samples identically distributed with this law. The objective of this study is to retrieve the dynamics of the system by predicting the future $(\underline{\mathbf{x}}_{t+H:t+1})$ given the past $(\underline{\mathbf{x}}_{t:t-l+1})$.

6.2.2 Data as Demonstrator

We decide in the following to focus on the recursive approach by using a single model that evolves over time by composition. The recursive approach brings the simplicity of implementation contrary to the multi-output approach, which could describe a non-interpretable complex model. We want to develop a model that takes advantage of this long-term prediction. We want a coherent and predictive model that is faithful to the underlying data-generating process. For this, an intuitive approach consists of solving the optimization problem as follows:

Given the nonlinear relationships and complex interactions between the input features, we introduce the expansion $\Phi(\underline{\mathbf{x}}_{t:t-l+1})$ of $\underline{\mathbf{x}}_{t:t-l+1}$ where $\Phi : \mathbb{R}^{pl+1} \to \mathbb{R}^{p_\Phi}$ and $p_\Phi \geq pl + 1$ is the number of variables after expansion. This function allows us to capture nonlinear relationships within the data. Various expansions can be considered including polynomial, sinusoidal, and other nonlinear transformations, but we impose that the first row of Φ is 1 encoding the intercept and the last pl rows contain the identity:

$$\Phi_{(i-1)p+j+1}(\underline{\mathbf{x}}_{t:t-l+1}) = x^{(j)}_{t-i+1}$$

for $i \in \{1, \dots, l\}$ and $j \in \{1, \dots, p\}$.

Let us define the $(pl + 1) \times p_\Phi$ matrix $\mathcal{C}$ such that, for a given t,

$$\hat{\underline{\mathbf{x}}}_{t+1:t-l+2} = \mathcal{C}\Phi(\underline{\mathbf{x}}_{t:t-l+1})$$

computes the one-step prediction. The matrix is the companion form as defined in equation 5.6. The first row of $\mathcal{C}$ is imposed as there $\mathcal{C}_{1,1} = 1$ and $\mathcal{C}_{1,j} = 0$ for $j > 1$. In addition, the last $p(l-1)$ rows of $\mathcal{C}$ are also imposed: all entries are zero except $\mathcal{C}_{ip+j,ip+j} = 1$ for $j \in \{1, \dots, p\}$ and $i \in \{1, \dots, l-1\}$. This ensures that the vector $\underline{\mathbf{x}}_{t:t+2-l}$ is predicted without any error. The row 2 to $p + 1$ of the matrix $\mathcal{C}$ contain the coefficients of the predictive equations

$$\hat{x}^{(j)}_{t+1} = \sum_{i=1}^{p_\Phi} \mathcal{C}_{j,i}\Phi_i(\underline{\mathbf{x}}_{t:t-l+1})$$

for $j \in \{1, \dots, p\}$. For the sake of simplicity of notation, we denote by M the set of such matrices with $p_\Phi p$ free entries.

Intuitively, a multi-step ahead predictor could be obtained by solving the following minimization problem:

$$\hat{\mathcal{C}}_\Phi = \underset{\mathcal{C}_\Phi \in M}{\operatorname{argmin}} \sum_{\delta=1}^{H} \mathbb{E}_{\mathcal{D}}\big[\|\underline{\mathbf{x}}_{l+\delta:\delta+1} - (\mathcal{C}_\Phi\Phi)^\delta(\underline{\mathbf{x}}_{l:1})\|^2\big],$$

or more exactly, its empirical version defined in terms of the training data set $\mathcal{D}_N$ as

$$\hat{\mathcal{C}}_\Phi = \underset{\mathcal{C}_\Phi \in M}{\operatorname{argmin}} \sum_{\delta=1}^{H} \sum_{t=l}^{T-\delta} \sum_{i=1}^{N} \|\underline{\mathbf{x}}_{i,t+\delta:t+1+\delta-l} - (\mathcal{C}_\Phi \Phi)^\delta(\underline{\mathbf{x}}_{i,t:t-l+1})\|^2, \tag{6.1}$$

where $\underline{\mathbf{x}}_{i,t:t-l+1}$ is a function of $\mathbf{z}_{i,t} \in \mathcal{D}_N$. Thus, the matrix $\hat{\mathcal{C}}_\Phi$ would contain the coefficients of the learned model estimating the underlying system generating the data and would be not only interpretable but coherent across future predictions. The issue here is that this problem is non-convex, so the determination of the global minimum is highly non-trivial and requires complex computation or approximations. As said before, the direct approach allows us to overcome this difficulty by estimating a matrix for each prediction horizon $\delta \in \{1, \dots, H\}$:

$$\hat{\mathcal{C}}_\Phi^{(\delta)} = \underset{\mathcal{C}_\Phi \in M^\delta}{\operatorname{argmin}} \sum_{t=l}^{T-\delta} \sum_{i=1}^{N} \|\underline{\mathbf{x}}_{i,t+\delta:t+1+\delta-l} - \mathcal{C}_\Phi \Phi(\underline{\mathbf{x}}_{i,t:t-l+1})\|^2, \tag{6.2}$$

where M^δ is the set of $pl \times p_\Phi$ matrices whose last $p(l-\delta)_+$ rows are imposed: all entries are zero except

$$\mathcal{C}_{ip+j,(i+\delta-1)p+j} = 1$$

for $j \in \{1, \dots, p\}$ and $i \in \{1, \dots, (l-\delta)_+\}$. This approach is limited because we neglect the statistical dependence and the coherence that exist in the underlying process.

The intuition that we want to exploit here is to use a recursive framework that allows us to compute an approximation of the model $\hat{\mathcal{C}}_\Phi^{(\delta)}$ in Eq. (6.2).

We would start by learning the one-step forecasting matrix

$$\hat{\mathcal{C}}_\Phi^{(1)} = \underset{\mathcal{C}_\Phi \in M}{\operatorname{argmin}} \sum_{t=l}^{T-1} \sum_{i=1}^{N} \|\underline{\mathbf{x}}_{i,t+1:t-l+2} - \mathcal{C}_\Phi \Phi(\underline{\mathbf{x}}_{i,t:t-l+1})\|^2$$

and then use this matrix to learn a two-step forecasting model

$$\hat{\mathcal{C}}_\Phi^{(2)} = \hat{\mathcal{C}}_\Phi^{(2,1)} \Phi \hat{\mathcal{C}}_\Phi^{(1)}$$

in an iterative way with

$$\hat{\mathcal{C}}_\Phi^{(2,1)} = \underset{\mathcal{C}_\Phi \in M}{\operatorname{argmin}} \sum_{t=l}^{T-2} \sum_{i=1}^{N} \|\underline{\mathbf{x}}_{i,t+2:t-l+3} - \mathcal{C}_\Phi \Phi(\hat{\mathcal{C}}^{(1)} \Phi(\underline{\mathbf{x}}_{i,t:t-l+1}))\|^2$$

and so on. The drawback here is that we still have a direct approach with different models, even though there is some recursion as we use previous models.
To overcome this limitation, we take advantage of the Data as Demonstrator (DaD) algorithm, an iterative algorithm allowing to use of the following recursive strategy: the first step of the DaD consists of learning a one-step model $\hat{\mathcal{C}}_\Phi^{(1)}$ as previously and predicting the future trajectory up to a horizon H by composing the learned model, $\{\underline{\hat{\mathbf{x}}}_{t+1:t-l+2}, \underline{\hat{\mathbf{x}}}_{t+2:t-l+3}, \dots, \underline{\hat{\mathbf{x}}}_{t+H:t-l+H-1}\} = \{\hat{\mathcal{C}}_\Phi^{(1)} \Phi(\underline{\mathbf{x}}_{t:t-l+1}), (\hat{\mathcal{C}}_\Phi^{(1)} \Phi)^2(\underline{\mathbf{x}}_{t:t-l+1}), \dots, (\hat{\mathcal{C}}_\Phi^{(1)} \Phi)^H(\underline{\mathbf{x}}_{t:t-l+1})\}$.
The main idea behind this algorithm is to reuse the predictions to artificially add training data and improve the H-step ahead prediction. As described in Figure 6.1, the intuition is to correct the predicted trajectory at each step by creating training pairs composed of the prediction and

the real labels. The model will learn a model that *corrects* its mistakes by returning to the true trajectory. Thus, at each iteration k, we have an augmented optimization problem containing the real data and also the previously predicted trajectories. For the sake of simplicity, we write the optimization problem up to step $T - H$ instead of $T - \delta$. We write the problem as follows: for $k \geq 0$,

$$\hat{\mathcal{C}}_{\Phi,k} = \underset{\mathcal{C}_\Phi \in M}{\operatorname{argmin}} \sum_{j=0}^{k} \ell_j(\mathcal{C}_\Phi), \tag{6.3}$$

where

$$\ell_0(\mathcal{C}_\Phi) = \frac{1}{T-H-l+1} \sum_{t=l}^{T-H} \frac{1}{N} \sum_{i=1}^{N} \|\underline{\mathbf{x}}_{i,t+1:t-l+2} - \mathcal{C}_\Phi \Phi(\underline{\mathbf{x}}_{i,t:t-l+1})\|^2 \tag{6.4}$$

is the one-step loss using the real data, and for $j \geq 1$,

$$\ell_j(\mathcal{C}_\Phi) = \frac{1}{L-l} \sum_{t=l+1}^{L} \frac{1}{N} \sum_{i=1}^{N} \frac{1}{M_t} \sum_{m=1}^{M_t} \|\underline{\mathbf{x}}_{i,t+1:t-l+2} - \mathcal{C}_\Phi \Phi(\hat{\underline{\mathbf{x}}}^{(j,m)}_{i,t:t-l+1})\|^2 \tag{6.5}$$

is the one-step loss evaluated on predictions generated by $\hat{\mathcal{C}}_{\Phi,j-1}$. Indeed, at iteration j, for each trajectory i and time step $t = l+1, \ldots, L$, we add M_t predictors $\hat{\underline{\mathbf{x}}}^{(j,m)}_{i,t:t-l+1}$ of $\underline{\mathbf{x}}_{i,t:t-l+1}$, $m = 1, \ldots, M_t$. In the original approach developed by Venkatraman et al. (2015), the enrichment of the data is done in the following way: $L = H$, $M_t = 1$, $\hat{\underline{\mathbf{x}}}^{(j,1)}_{i,t:t-l+1} = (\hat{\mathcal{C}}_{\Phi,j-1}\Phi)^{t-l}\underline{\mathbf{x}}_{i,l:1}$ for $t = l+1, \ldots, L$ (assuming $H > l$; we have $l = 1$ in (Venkatraman et al., 2015)). In section 6.3, we introduce new enrichment procedures.

Note that Eq. (6.3) is a simple least-squares problem that can be regularized using ℓ_2 or ℓ_1 penalty. Because we want to retrieve the underlying properties of the dynamical system, we use the ℓ_1 penalty to promote sparsity, as only a few variables at specific time lags have an influence. This constraint simplifies the model by focusing on the most critical variables, making it easier to interpret.

Once we have reached the number of iterations K set beforehand, we obtain several candidates to find the coefficients governing the dynamics of the time series. The selection is made through a validation data set on which we choose the optimal one. Finally, we use a test data set to judge the accuracy of the multi-step ahead prediction.

6.2.3 Guarantees and limitations

DaD is inspired by the DAgger algorithm, an interactive imitation learning algorithm. The paper of Venkatraman et al. (2015) states guarantees in the form of two theorems that are extensions of Ross et al. (2011) work. A first theorem guarantees the model's ability to be good on its own trajectory, and a second one links the multi-step loss to the number of iterations. However, we can see in our simulations that the multi-step loss of the model, defined as

$$\sum_{\delta=1}^{H} \sum_{t=l}^{T-\delta} \sum_{i=1}^{N} \|\underline{\mathbf{x}}_{i,t+\delta:t+1+\delta-l} - (\mathcal{C}_\Phi \Phi)^\delta(\underline{\mathbf{x}}_{i,t:t-l+1})\|^2, \tag{6.6}$$

deviates a lot during the first iterations of the described approach. These deviations increase with the length of the horizon to predict. Indeed, we can see in Figure 6.2 that the DaD model

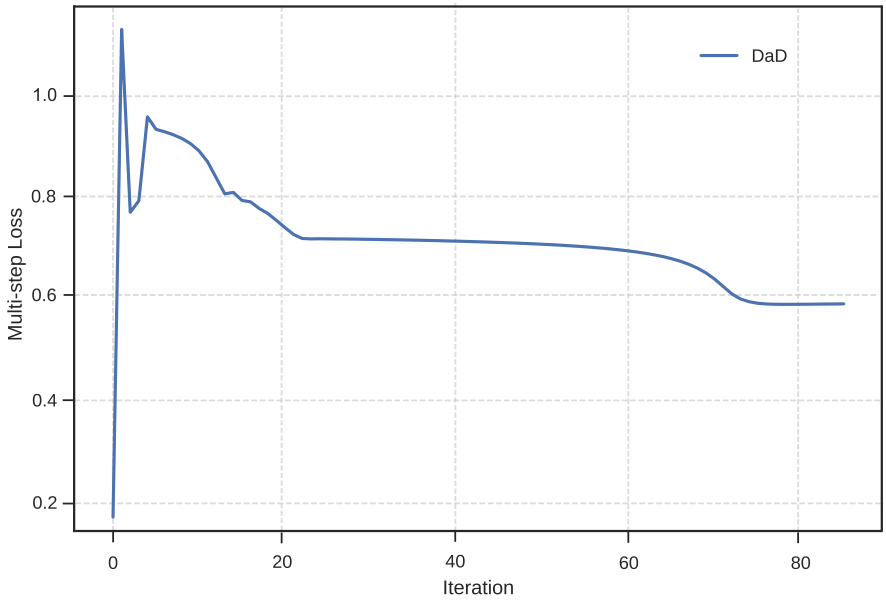


Figure 6.2: This figure illustrates the losses of the original DaD algorithm on the Cartpole dataset with respect to the DAgger iteration. The parameters H and T are set at 30 and 50, respectively.

suffers during the first iterations of a poor ability to predict in the long run when adding corrections. We note that the DaD's deviation is significant enough to prevent it from making adequate corrections to achieve a low loss.

In addition, the algorithm does not allow the exploitation of the data in the best way because the training data set is only increased from a single point of the initial trajectory. Moreover, large-horizon forecasts inherently involve higher uncertainty due to the potential influence of various unforeseen factors. These factors can cause the predictions to diverge from the actual values, leading to poor performance.

To overcome these limitations, we have developed several methods described in the following section.

6.3 Methodology

Motivation This work aims to infer the dynamic model that drives the data generation from available samples. We focus on learning a long-term coherent model, meaning that for a stationary system, the model should accurately predict its future state at any given number of steps from any point in time. To achieve this, we investigate and extend the recursive approach to have a coherent model, particularly the DaD algorithm, to enhance long-term prediction abilities. Indeed, using an interpretable and coherent model in the long term allows us to obtain higher reliability for the prediction. In this section, we propose an original method inspired by DaD, which can process continuous or categorical temporal multidimensional time series data.

data-set augmentation We study different novel approaches to improve the approach developed in DaD and better use the training data. These methods are similar in that they aim to be coherent in long-term prediction and to take advantage of the whole data set. In the following, we explain the details of these methods. Our framework differs on the predictive part of the trajectory, allowing us to increase the data set to do better H-step ahead forecasting.

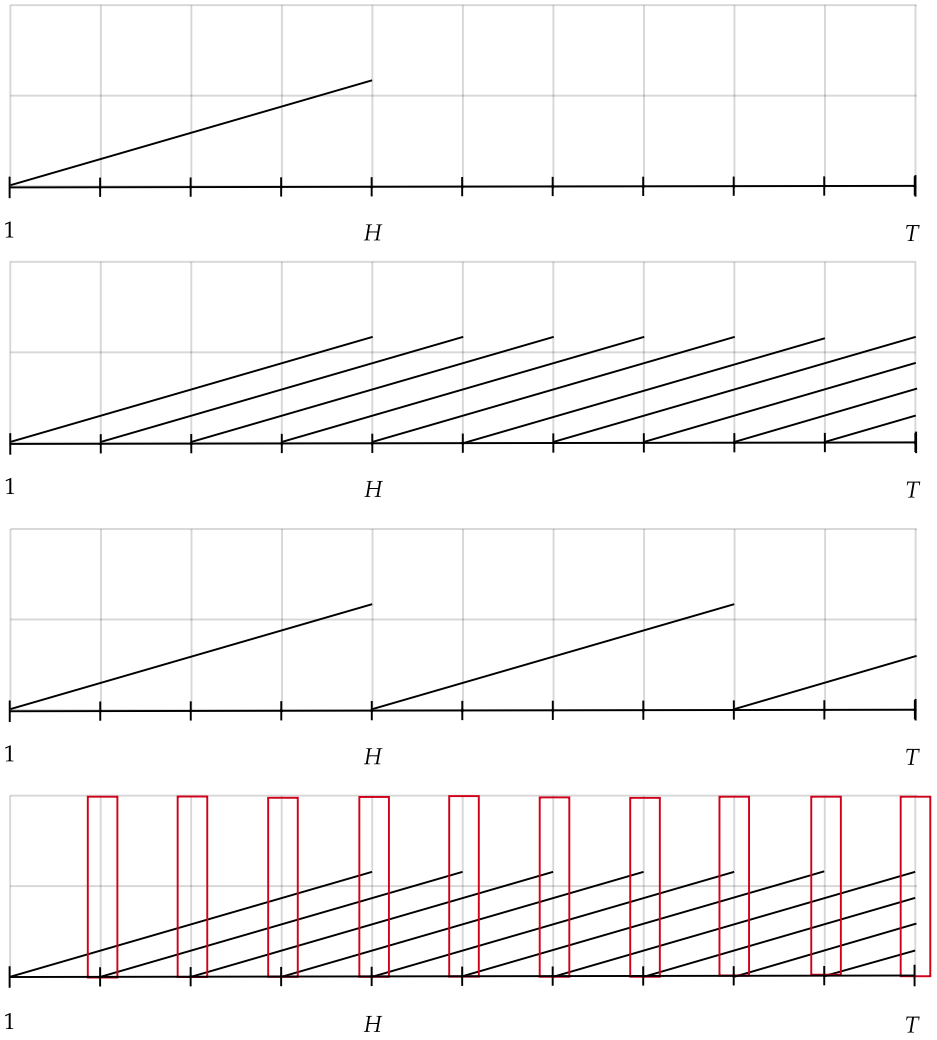


Figure 6.3: From top to bottom: DaD method, DaD_R, i.e., the optimized method with several restarts, DaD_sub (subsampling), and DaD_agg (aggregation). This figure illustrates four methods of data augmentation used during the training process (for $l = 1$). The first diagram illustrates the addition of a single predicted trajectory of length H, starting from the initial time step. The second diagram illustrates the addition of trajectories predicted for each time step in the sequence. The last two diagrams illustrate techniques for reducing the number of additional samples by applying aggregation and sub-sampling approaches, respectively.

DaD_R The objective of this new setting is to learn a one-step model similar to the DaD algorithm. Then, the data-set augmentation will be done using not one but multiple trajectories.

In fact, we will predict several h-step ahead trajectories where $h \leq H$. We will use the learned matrix and predict trajectories starting at each time step $t < T$ and up to the horizon $\min\{T-t, H\}$. The idea is to augment the data set with all possible predictions using the learned model.

Thus, at each time step, we can have several forecasts. For each t, we predict $\hat{\underline{\mathbf{x}}}_{t:t-l+1}$ with $\mathcal{C}_\Phi\Phi(\underline{\mathbf{x}}_{t-1:t-l})$ up to $(\mathcal{C}_\Phi\Phi)^H(\underline{\mathbf{x}}_{t-H:t-H-l+1})$ for $t > H$. More exactly, at each iteration $j \geq 1$, for $t \geq l+1$, we have the following M_t predictions:

$$\hat{\underline{\mathbf{x}}}^{(j,m)}_{i,t:t-l+1} = (\hat{\mathcal{C}}_{\Phi,j-1}\Phi)^m(\hat{\underline{\mathbf{x}}}_{i,t-m:t-l-m+1}), \text{ for } m \in \{1, \dots, M_t\}.$$

In this setting schematized in the second position of figure 6.3, we have $L = T - H$ and $M_t = \min\{t-l, H\}$.

Although this approach allows the best use of the data set and to learn the matrix, it has practical limitations. Indeed, at each DAgger iteration, we add $T-2$ trajectories with sizes between 1 and H. This requires a large amount of memory usage, especially if T, H, and p are large. The approach must, therefore, be adapted and simplified.

For this purpose, we add two ingredients that allow us to make the best use of these different trajectories: sub-sampling and increasing horizons. First, sub-sampling the trajectories will reduce the computational burden while still learning a coherent and faithful underlying generative model. Second, we have observed that increasing slowly the horizon of predictions yields better predictive results. The corresponding ablation study is available in section 6.4.4.2.

DaD_agg For the first method, we decide to aggregate the predictions by averaging the available predictors. We impose each estimator to be good on several time steps. As shown in Figure 6.3, the method aggregates the different predictions in the red rectangle for specific time steps. Thus, at each iteration $j \geq 1$, for $t \geq l+1$, we have the following prediction:

$$\hat{\underline{\mathbf{x}}}^{(j,1)}_{i,t:t-l+1} = \frac{1}{\min(t-l, H)} \sum_{m=1}^{\min(t-l,H)} (\hat{\mathcal{C}}_{\Phi,j-1}\Phi)^m(\underline{\mathbf{x}}_{i,t-m:t-l-m+1}). \tag{6.7}$$

In this setting, we have $L = T - H$ and $M_t = 1$.

DaD_sub For the first ingredient, sub-sampling, we predict from a starting point and compose the learned matrix until the predefined maximal horizon H restarts at one step, and then we compose again after reaching it. Here, we use a multi-step ahead forecast, and we decided to subsample so that we have multiple trajectories that do not overlap, as shown in figure 4.6. Hence, at each iteration $j \geq 1$, for $t \geq l+1$, we have the following prediction:

$$\hat{\underline{\mathbf{x}}}^{(j,1)}_{i,t:t-l+1} = (\hat{\mathcal{C}}_{\Phi,j-1}\Phi)^{t-\mathcal{H}(t)}(\underline{\mathbf{x}}_{i,\mathcal{H}(t):\mathcal{H}(t)+1-l}),$$

where $\mathcal{H}(t) = \max(l, H\lfloor\frac{t-1}{H}\rfloor)$. In this subsampling method, $L = T - H$ and $M_t = 1$.

DaD_sub$^+$ and DaD_agg$^+$ For the second ingredient, we develop an approach that emphasizes the progressive nature of learning. Indeed, the model first learns to predict in the short term at a prediction horizon h_1. Once the learning is completed, i.e., when no

improvement can be detected during the DAgger iterations, we increase the prediction horizon until the final horizon H is reached. Thus, this approach allows the model to avoid learning a task that is too difficult, i.e., learning in the short term and at the same time in the long term with a very large horizon.

More exactly, at each DAgger iteration, we test whether the score improves, and we set a threshold β. Suppose the score does not improve after a number of iterations k_s. In that case, we update the prediction horizon by a fixed number of steps S and reuse the previously augmented data set (before the k_s iterations). We write $\mathcal{D}^{(k)}$ the augmented training data set used at the iteration k. The problem is as follows: if

$$\begin{aligned}\frac{1}{h_k(T-H-l+1)N}\sum_{\delta=1}^{h_k}\sum_{t=l}^{T-H}\sum_{i=1}^{N}\Big[&\|\underline{\mathbf{x}}_{i,t+\delta:t-l+\delta+1}-(\mathcal{C}_{\Phi,k-1}\Phi)^{\delta}(\underline{\mathbf{x}}_{i,t:t-l+1})\|^2\\ &-\|\underline{\mathbf{x}}_{i,t+\delta:t-l+\delta+1}-(\mathcal{C}_{\Phi,k}\Phi)^{\delta}(\underline{\mathbf{x}}_{i,t:t-l+1})\|^2\Big]\leq\beta,\end{aligned} \tag{6.8}$$

then $h_{k+1}=\min(h_k+S,H)$ and $\mathcal{D}^{(k+1)}=\mathcal{D}^{(k+1-k_s)}$.

Validation After reaching the number of DAgger iterations previously fixed, we obtain a set of candidate matrices for each method. To determine which matrix is the most reliable and coherent, we use a validation set $\mathcal{D}^{\mathrm{v}}$ of N_{v} trajectories and denote by $\underline{\mathbf{x}}^{\mathrm{v}}_{i,t:t-l+1}$ the elements of the data-set. The selected model is the learned model with the lowest validation loss:

$$\hat{\mathcal{C}}_{\Phi}=\underset{\mathcal{C}\in\{\hat{\mathcal{C}}_1,\ldots,\hat{\mathcal{C}}_K\}}{\operatorname{argmin}}\sum_{\delta=1}^{H}\sum_{t=l}^{T-H}\sum_{i=1}^{N_{\mathrm{v}}}\|\underline{\mathbf{x}}^{\mathrm{v}}_{i,t+\delta:t-l+\delta+1}-(\mathcal{C}\Phi)^{\delta}(\underline{\mathbf{x}}^{\mathrm{v}}_{i,t:t-l+1})\|^2.$$

Computation These methods differ in time and space complexity. In fact, at each iteration and for each trajectory: 1) DaD needs to store an additional trajectory of length H, hence H samples; 2) the method DaD_R constructs $T-1$ trajectories, with $T-H$ trajectories of length H and the remaining trajectories of length $T-H$ to 1, which makes $(T-H)H+\frac{(H-1)H}{2}$ samples; 3) the method DaD_sub store $T-1$ samples.

6.3.0.1 Theoretical analysis

DaD is an iterative algorithm that learns a matrix $\hat{\mathcal{C}}_{\Phi}$ from the training data to perform multi-step prediction. The learned matrix is used to simulate trajectories, allowing us to judge the quality of the learning and correct it. The matrix at the following iterations is learned from the previous data: on the one hand, the real trajectory with the pairs allowing to learn the one-step prediction, and on the other hand, the pairs formed from the values of the simulated trajectories associated with the values given by the real trajectory. Thus, the matrix $\hat{\mathcal{C}}_{\Phi}$, in addition to learning on the real trajectory, has to make 'corrections' that allow us to learn the underlying model.

This approach can be interpreted as an interactive imitation learning approach and, more particularly, as the DAgger algorithm developed by Ross et al. (2011). Indeed, as described by Venkatraman et al. (2015), we can rewrite our problem in this framework and thus take advantage of the theoretical guarantees we will develop further.

More precisely, note that at iteration k, the algorithm learns a model

$$\hat{\mathcal{C}}_{\Phi,k} = \text{argmin}_{\mathcal{C}_\Phi} \sum_{j=0}^{k} \ell_j(\mathcal{C}_\Phi)$$

from $\mathcal{D}^{(k)}$. We would be interested in controlling the loss of the coherent model $\ell_j(\hat{\mathcal{C}}_{\Phi,j-1})$, i.e., the loss of a model where the same matrix $\hat{\mathcal{C}}_{\Phi,j-1}$ is used to build the trajectories and to make the one-step prediction.

Thanks to the Follow-The-Leader principle of our algorithm, control of the coherent loss by the loss truly minimized is possible. Indeed, using the fact that our losses ℓ_j are σ_j strongly convex, we can prove the following theorem using Shalev-Shwartz & Kakade (2008) analysis.

Theorem 6.3.1. *For any $K \geq 1$,*

$$\min_{1\leq j\leq K} \ell_j(\hat{\mathcal{C}}_{\Phi,j-1}) \leq \min_{\mathcal{C}} \frac{1}{K}\sum_{j=0}^{K} \ell_j(\mathcal{C}) + \frac{\log(K+1)}{2K} \frac{\sup_{j\leq K} \|\partial \ell_j(\hat{\mathcal{C}_\Phi}_{j-1})\|^2}{\min_{j\leq K} \sigma_j},$$

where $\partial \ell_j$ is a subgradient of ℓ_j.

Proof. The proof lies in the fact that our framework, written for $k \geq 0$,

$$\hat{\mathcal{C}}_{\Phi,k} = \underset{\mathcal{C}\in M}{\text{argmin}} \sum_{j=0}^{k} \ell_j(\mathcal{C})$$

is a Follow-The-Leader (FTL) algorithm. Indeed, following Shalev-Shwartz & Kakade (2008) analysis, we have the following upper-bound:

$$\sum_{j=0}^{K} \ell_j(\hat{\mathcal{C}}_{\Phi,j-1}) - \min_{\mathcal{C}\in M} \sum_{j=0}^{K} \ell_j(\mathcal{C}) \leq \frac{1}{2}\sum_{j=0}^{K} \frac{\|\partial \ell_j(\hat{\mathcal{C}}_{\Phi,j-1})\|^2}{\sum_{i=1}^{j}\sigma_i}.$$

Without loss of generality, let $\hat{\mathcal{C}}_{\Phi,-1} = 0$. Then,

$$\begin{aligned}
\min_{1\leq j\leq K} \ell_j(\hat{\mathcal{C}}_{\Phi,j-1}) &\leq \min_{\mathcal{C}\in M} \frac{1}{K}\sum_{j=0}^{K} \ell_j(\mathcal{C}) - \frac{1}{K}\ell_0(0) + \frac{1}{2K}\sum_{j=0}^{K} \frac{\|\partial \ell_j(\hat{\mathcal{C}}_{\Phi,j-1})\|^2}{\sum_{i=1}^{j}\sigma_i} && \text{(Definition of minimum)} \\
&\leq \min_{\mathcal{C}\in M} \frac{1}{K}\sum_{j=0}^{K} \ell_j(\mathcal{C}) + \frac{1}{2K}\sum_{j=0}^{K} \frac{\|\partial \ell_j(\hat{\mathcal{C}}_{\Phi,j-1})\|^2}{\sum_{i=1}^{j}\sigma_i} && \text{(Non-negativity of loss)} \\
&\leq \min_{\mathcal{C}\in M} \frac{1}{K}\sum_{j=0}^{K} \ell_j(\mathcal{C}) + \frac{\log(K+1)}{2K} \frac{\sup_{j\leq K} \|\partial \ell_j(\hat{\mathcal{C}}_{\Phi,j-1})\|^2}{\min_{j\leq K} \sigma_j} && \text{(Upper-bound)}
\end{aligned}$$

■

Remark 6.3.2. *We still remain under this theorem by adding an L_1 penalty since the total loss is σ_j-strongly convex.*

Provided the gradient is upper bounded, and the σ_j are lower bounded, as we have observed in practice, this means that a small trained loss implies a good coherent model.

6.4 Application

In this section, we evaluate the capacity of our model to learn the system dynamics by studying its ability to predict at several steps up to a fixed horizon H and to capture the structure by analyzing the estimated coefficients. We compare it to multiple algorithms on synthetic and real-world data sets.

6.4.1 Evaluation metrics

The models are evaluated along two dimensions. The first dimension assesses their predictive capability, which refers to their accuracy in generating forecasts. The second analyzes the model's ability to capture the underlying mechanisms or relationships within the data.

6.4.1.1 Predictive analysis

To evaluate the predictive ability of our approach, we have trained the DaD, DaD_R iteratively, DaD_sub and DaD_sub$^+$, until we obtain the convergence of the loss on the validation set. For the incremental models, we set $h_1 = 1$, $\beta = 0.01$, $S = 1$ and $k_s = 10$. To compare the advantages of the methods using data enrichment, we also trained a one-step Linear Regression (method LR). Besides, we learned a model using the direct approach also with a Linear Regression determined by Eq. (6.2) (Direct method). Indeed, comparing the direct approach with the developed methods is useful for providing an upper bound on what might be achievable by any recursive model.
All models have multivariate outputs and predict at multiple steps. To evaluate them, we use the normalized root mean squared error (NRMSE):

$$\frac{\sqrt{\sum_t \|\hat{\mathbf{x}}_t - \mathbf{x}_t\|_2^2}}{\sqrt{\sum_t \|\mathbf{x}_t\|_2^2}}. \tag{6.9}$$

This metric allows us to obtain a performance measure of a multi-step forecasting model. Indeed, for each horizon $h < H$, we measure the difference between the predictions at h steps and the true values. We normalize to constrain the model to predict well for all variables.

6.4.1.2 Interpretability analysis

When the data comes from a discrete dynamical system, we are interested in assessing the extent to which a model can identify true relationships between variables and the effect of multicollinearity. Since we study known systems in this section, we evaluate the model's performance by comparing the variables found with the *true* coefficients. More precisely, we define the sets T and P of the true variables and the predicted variables, respectively. To take into account the collinearity in the data, we define the correlation matrix *corr* where the element corr$[i, j]$ represents the correlation between $x_{T:1}^{(i)}$ and $x_{T:1}^{(j)}$ for all $i, j \in \{1, \dots, n\}$. We introduce two other sets

$$\mathrm{T}^+ = \{i | j \in \mathrm{T}, |\mathrm{corr}[i, j]| \geq \alpha\}$$

and

$$\mathrm{P}^+ = \{i | j \in \mathrm{P}, |\mathrm{corr}[i, j]| \geq \alpha\}$$

which are, respectively, the variables correlated more than a threshold $\alpha \in [0, 1]$ with the true variables and the predicted variables. In the following, we show the results for $\alpha = 0.9$ (an in-depth analysis of the effect of this parameter is given in the appendix B).

To measure interpretability, we first introduce metrics derived from the confusion matrix metrics in table 6.1 where |E| represents the cardinal of the set E and $\overline{\mathrm{E}}$ is the complementary set of E.

Metric	**Definition**
True Positives (TP)	$\lvert \mathrm{P} \cap \mathrm{T}^+ \rvert$
False Positives (FP)	$\lvert \mathrm{P} \cap \overline{\mathrm{T}^+} \rvert$
True Negatives (TN)	$\lvert \overline{\mathrm{T}} \cap \overline{\mathrm{P}^+} \rvert$
False Negatives (FN)	$\lvert \mathrm{T} \cap \overline{\mathrm{P}^+} \rvert$

Table 6.1: Confusion Matrix Metrics

These metrics, including numbers of True Positives (TP), False Positives (FP), True Negatives (TN), and False Negatives(FN), allow us to define various metrics providing insight into the model's interpretability performance, such as the F_2-score, Sparsity score, Recall and Precision presented in table 6.2.

Metric	**Definition**
F_2 Score	$\frac{(1+2^2)\cdot TP}{(1+2^2)\cdot TP+2^2\cdot FN+FP}$
Sparsity	$\frac{\lvert \mathrm{P} \rvert}{\lvert \mathrm{T} \rvert}$
Recall	$\frac{\lvert \mathrm{T} \cap \mathrm{P}^+ \rvert}{\lvert \mathrm{T} \rvert}$
Precision	$\frac{\lvert \mathrm{P} \cap \mathrm{T}^+ \rvert}{\lvert \mathrm{P} \rvert}$

Table 6.2: Evaluation Metrics: F2-score, sparsity, recall, and precision

The appendix B evaluates additional metrics further. Other factors, such as noise magnitude, horizon H, and data-set size, can influence the model's ability to recover the true causal coefficients, so they are considered in the analysis.

In this study, we compare our approach with a discrete SINDy (Brunton et al., 2016), k-Best, Recursive Feature Elimination and SelectFI (Pedregosa et al., 2011).

6.4.2 Benchmark Methods

This paragraph outlines the comparative methodologies used in our study:

- SINDy (Sparse Identification of Nonlinear Dynamics) with Lasso: This method is a multivariate feature selection algorithm. It operates by selecting the most relevant features based on Lasso regularization after a nonlinear expansion of the variables. In our study, the SINDy algorithm with Lasso identifies the sparsest set of variables that can accurately represent the system's dynamics. In our study, we determine the optimal regularization parameters by cross-validation.

- kBest with f-regression: This method is a univariate feature selection algorithm. It selects the k best features based on univariate statistical tests. It can be seen as a preprocessing step to an estimator. In this case, we use the f-regression function as the score function, which is used for regression tasks. The f-regression function computes the correlation between each regressor and the target and converts it into an F score, then to a p-value. In our study, we identify the optimal parameter k by cross-validation.

- SelectFromModel with Random Forest: SelectFromModel is a meta-transformer that can be used along with any estimator that assigns importance to features, either through a coef_ attribute or through a feature_importances_ attribute. In this case, we use a Random Forest Regressor as an estimator. The features are selected based on their importance weights, with the less important ones removed. Different criteria were used, such as the mean and the median.

- Recursive Feature Elimination (RFE) with Gradient Boosting: RFE is a feature selection method that fits a model and removes the weakest features until the specified number of features is reached. Features are ranked by the model's coef_ or feature_importances_ attributes, and by recursively eliminating a small number of features per loop, RFE attempts to eliminate dependencies and collinearity that may exist in the model. In this case, we use a Gradient Boosting Regressor as an estimator. In our study, we identify the optimal number of features to remove by cross-validation.

These last three approaches are all implemented within the scikit-learn library (Pedregosa et al., 2011).

6.4.3 Data-sets

To assess our method, we rely on synthetic data sets with known ground truth. This controlled setting enables a rigorous evaluation of the method's performance under well-defined conditions. Subsequently, the method's performance is further validated using a real-world data set. Details on the data sets are given below.

6.4.3.1 Synthetic data-sets

We test our methods in two different synthetic settings:

- The Ordinary Differential Equations (ODEs) data was derived by implementing the fourth-order Runge-Kutta numerical method to solve the differential equations. This generated a time series from which a subsample was extracted to constitute our final data set. It is important to mention that noise is incorporated during the data generation process. Specifically, while the ODE is being solved using the Runge-Kutta method, a noise component is added to the system at each iteration.
  The Cartpole, the Pendulum, the Double Pendulum, the Spring Pendulum, and the System Mass Spring Pendulum are generated in this way. The following are the parameters used for data generation: $N = 90$ trajectories have been generated, which are time series with different initial conditions of length $T = 200$. Among these trajectories, 30 are used for training, 20 for validation, and 40 for testing. The prediction horizon is $H = 70$.

- The data for the Discrete Dynamical Systems was obtained by applying a first-order discretization to an Ordinary Differential Equation for both the Lotka-Volterra and SIR models. The following parameters are used for data generation: $N = 80$ trajectories have been generated, which are time series with different initial conditions. Among these trajectories, 30 are used for training, 20 for validation, and 20 for testing. The prediction horizon is $H = 10$.

In the first one, we observe discrete dynamical systems associated with delayed Lotka-Volterra and SIR equations. In this setting, we use a polynomial expansion of degree three as Φ, and the true dynamical systems can be exactly expanded onto Φ. In the second one, we observe discrete time samples of solutions of non-linear ODEs (Cartpole, single pendulum, mass pendulum combined with a string, mass attached to a pendulum). As mentioned above, a true linear discrete system exists in expanded variables only in the first setting. The section B.1 of the appendix gives more details on the data generation process.

6.4.3.2 Real-world data-set

The data set presented in this study includes real-world wind turbine energy production data collected over a two-year period from five wind farms operated by the private company Zéphyr ENR (Bouche et al., 2022; Dupré et al., 2020b) and forecasts data set from the European Centre for Medium-Range Weather Forecast (2024) (ECMWF). The five wind farms considered are located in France in Parc de Bonneval, Moulin de Pierre, Parc de Beaumont, Parc de la Renardière, and Parc de la Vènerie.

The data set provided by Zéphyr ENR contains sensor measurements from the wind turbines. These measurements, known as in-situ variables, include variables such as wind speed and Power output. The variables are described in table 6.3.

Variable	**Unit**
Wind speed	ms^{-1}
Power output	kW
Wind direction	Degrees
Temperature	Celcius degree

Table 6.3: In-situ variables

The ECMWF provides a data set of global forecasts generated by numerical weather prediction (NWP) models. Following (Dupré et al., 2020a) description, this data set includes forecasts for the next day, containing 47 atmospheric variables extracted twice daily detailed in table 6.4. These variables variables describe the boundary layer, winds, and temperature in the lower troposphere. Additionally, ECMWF forecasts have a spatial resolution of approximately 16 kilometers. To account for specific farm locations, missing data points are interpolated linearly from the four nearest grid points.

The central focus of this study is to model the time series wind turbine power output based on other measurements and forecasts. We use $N = 30$ time series of length $T = 28$, of lag $l = 2$, and a horizon $H = 15$. Among these trajectories, 10 are used for training, 10 for validation, and 10 for testing.

Variable type	Altitude or pressure level	Variable	Unit
Surface	10m/100m	Zonal wind speed	ms^{-1}
		Meridional wind speed	ms^{-1}
	2m	Temperature	K
		Dew point temperature	K
		Skin temperature	K
		Mean sea level pressure	Pa
		Surface pressure	Pa
		Surface latent heat flux	Jm^{-2}
		Surface sensible heat flux	Jm^{-2}
		Boundary layer dissipation	Jm^{-2}
		Boundary layer height	m
Altitude	1000/925/850/700/500	Zonal wind speed	ms^{-1}
		Meridional wind speed	ms^{-1}
		Geopotential height	m^2s^{-2}
		Divergence	s^{-1}
		Vorticity	s^{-1}
		Temperature	K
Computed	10m/100m	Norm of wind speed	ms^{-1}
	10m to 925 hPa	Wind shear	ms^{-1}
		Temperature gradient	K

Table 6.4: ECMWF variables

6.4.4 Results

The results section first presents an ablation study to identify the optimal strategy among those developed in section 6.3. Subsequently, the results for discrete dynamical systems, ODE time series, and real-world data sets are detailed. All results are obtained in this section by averaging 50 runs over data generated with different initial conditions.

6.4.4.1 Training deviation

We observe, in the figure 6.4, that both DaD_sub and DaD_agg approaches also experience the deviation during the first iterations of the algorithm as seen in figure 6.2, but to a lesser degree. These methodologies allow us to make rapid adjustments and quickly reach good performance levels.

6.4.4.2 Ablation Study

An ablation study was carried out to compare the different methodologies developed in section 6.3 such as DaD, DaD_R, DaD_sub and DaD_sub$^+$. We conducted an analysis of the prediction loss in relation to varying noise levels and data-set sizes.

Our examination of noise levels, shown in Table 6.5, revealed that DaD_sub$^+$ and DaD_sub exhibit similar performance, significantly surpassing DaD and DaD_R. However, it was observed that DaD_sub outperforms the others in highly noisy settings at a level of 0.1, while DaD_sub$^+$'s performance deteriorates under these conditions. Upon further investigation, we

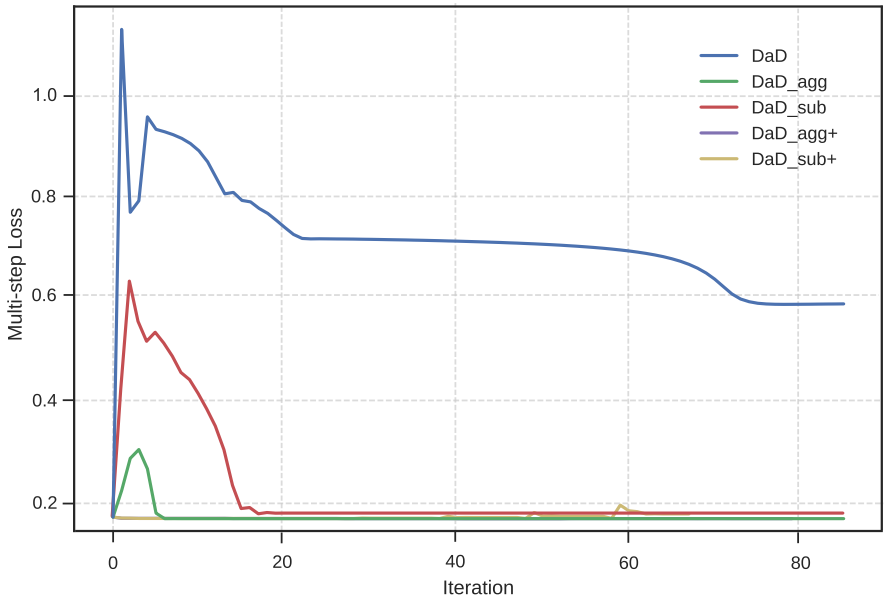


Figure 6.4: This figure illustrates the losses of the original DaD algorithm and the variant that we have developed on the Cartpole dataset with respect to the DAgger iteration. The parameters H and T are set at 30 and 50, respectively.

have seen that DaD_sub$^+$ performs poorly at a high noise level of 0.1 and a small data-set size of 2, contributing to the large standard deviation noted in Table 6.5.

Noise	DaD	DaD_R	DaD_sub	DaD_sub$^+$
0.01	0.00338±0.00182	0.51237±0.07712	0.0004±3e-05	**0.00037**±3e-05
0.03	0.04379±0.01166	0.54354±0.05304	0.00345±0.00011	**0.00248**±0.00018
0.07	0.12409±0.02027	0.49423±0.08779	0.01654±0.00156	**0.01292**±0.00128
0.1	0.15574±0.03546	0.42311±0.13289	**0.03586**±0.00495	0.2272±0.29191

Table 6.5: Ablation Study on Lotka-Volterra Data-set: Comparison of the prediction metric, NRMSE, for the approaches developed in this article and for different noise levels. The mean and standard deviation are computed over 50 runs.

Furthermore, the analysis of data-set size, as presented in Table 6.6, indicates that DaD_sub$^+$ consistently surpasses all other methods, irrespective of the data-set size.

Data-set size	DaD	DaD_R	DaD_sub	DaD_sub$^+$
2	0.08±0.085	0.505±0.045	0.011±0.019	**0.009**±0.278
5	0.091±0.076	0.522±0.047	0.01±0.016	**0.007**±0.012
15	0.08±0.053	0.497±0.158	0.009±0.014	**0.007**±0.045

Table 6.6: Ablation Study on Lotka-Volterra Data-set: Comparison of the prediction metric, NRMSE, for the approaches developed in this article and for different data-set sizes. The mean and standard deviation are computed over 50 runs.

This suggests that sub-sampling enhances the model's learning capability, and the incremental process helps improve predictions in every scenario except when dealing with strong noise and limited data availability. Therefore, DaD_sub$^+$ is the best performing method and will be the only one for which we will show results in the following experiments.

6.4.4.3 Discrete dynamical system

Table 6.7 and 6.8 summarize our results, obtained by averaging 50 runs over data generated with different initial conditions.

Regarding interpretability, which is well-defined here as there is an underlying linear dynamical system in expanded variables, we obtain strong performances across the diverse interpretability metrics described above. While it may not always attain the top position on all metrics, it is always ranked among the top two performers with performance close to the winner.

METRICS	METHODS				
	DaD_sub$^+$	SINDy	kBest	RFE	SelectFI
$\overline{F_2}$	**0.891**	<u>0.880</u>	0.823	0.815	0.667
$\overline{\text{Precision}}$	<u>0.683</u>	0.637	0.622	0.598	**0.759**
$\overline{\text{Recall}}$	<u>0.976</u>	**0.986**	0.880	0.893	0.507
$\overline{FP}$	<u>2.224</u>	2.704	9.200	4.910	**1.420**
$\overline{\text{Sparsity}}$	<u>1.993</u>	2.252	6.367	3.677	**1.460**
$\overline{\text{NRMSE}}$	**0.869**	1.009	/	0.997	<u>0.989</u>

Table 6.7: Evaluation on discrete Lotka-Volterra data-set with $\sigma = 0.07$, $N = 5$ and $H = 12$ and where $\overline{X}$ denotes the mean value of the metric X over 50 runs.

Additionally, our approach demonstrates the strongest predictive ability with the lowest NRMSE loss, showing effectiveness in making multi-step prediction. Similar results are obtained with the SIR (details are given in section B.2 of the appendix).

In this setting, we have observed that the best solution is consistently obtained when the algorithm considers a horizon of order 2 and that increasing the horizon does not yield much better results. This observation is consistent across different data-set sizes and noise levels. We think this is due to the same optimal solution, whatever the horizon.

6.4.4.4 ODE time series

In Table 6.9, we present a comparison of our forecasting approaches alongside alternative methods using the NRMSE metric. The results are obtained by performing multi-step forecasting on the test data set and by averaging all NRMSEs for each time step and variable.

Data-set Size	Metric	DaD_sub^+	SINDy	kBest	RFE	SelectFI
5	F2-score	**0.965**±0.007	0.959±0.005	0.587±0.004	0.543±0.006	0.585±0.001
	Precision	**0.884**±0.017	0.871±0.013	0.437±0.007	0.446±0.01	0.639±0.005
	Recall	**1.0**±0.0	0.999±0.001	0.663±0.002	0.529±0.016	0.429±0.003
	FP	1.507±0.562	2.116±0.328	12.567±1.185	2.58±0.207	**0.335**±0.077
	Sparsity	2.671±0.315	3.059±0.134	12.71±0.717	3.121±0.132	**2.135**±0.038
	NRMSE	/	5.048±0.779	**0.036**±0.068	0.578±0.004	0.794±0.01
10	F2-score	**0.962**±0.012	0.949±0.004	0.584±0.002	0.527±0.006	0.565±0.002
	Precision	**0.962**±0.012	0.949±0.004	0.584±0.002	0.527±0.006	0.565±0.002
	Recall	**1.0**±0.0	1.0±0.0	0.667±0.0	0.596±0.008	0.408±0.006
	FP	1.83±1.02	2.995±0.377	13.548±1.063	4.697±0.324	**0.188**±0.049
	Sparsity	2.86±0.582	3.533±0.192	13.322±0.638	4.339±0.122	**1.676**±0.021
	NRMSE	0.937±0.107	/	5.067±0.82	0.565±0.001	**0.785**±0.001
15	F2-score	**0.96**±0.014	0.948±0.005	0.58±0.003	0.518±0.007	0.556±0.001
	Precision	**0.873**±0.033	0.842±0.011	0.415±0.009	0.332±0.004	0.648±0.006
	Recall	**1.0**±0.0	1.0±0.0	0.667±0.0	0.608±0.014	0.391±0.003
	FP	2.027±1.104	3.185±0.38	15.062±1.781	5.075±0.078	**0.143**±0.044
	Sparsity	2.973±0.592	3.639±0.176	14.326±1.231	4.417±0.073	**1.457**±0.014
	NRMSE	**0.001**±0.001	0.001±0.002	4.911±0.824	0.564±0.001	0.78±0.003

Table 6.8: SIR Data-set: Comparison of the models over data-set sizes on interpretability metrics, F2-score, Precision, Recall, FP, Sparsity and prediction metric, NRMSE. The mean and standard deviation are computed over 50 runs. The winning method is shown in bold and the second is underlined.

Table 6.9: RNMSE for the different methods on ODE data sets

data-setS	METHODS					
	DaD	DaD_R	DaD_sub	DaD_sub$^+$	LR	Direct
Cartpole	0.338	0.338	0.338	**0.321**	0.338	0.306
Pendulum	0.274	0.273	0.278	**0.265**	0.274	0.136
Double Pendulum	0.626	0.565	0.624	**0.541**	0.615	0.503
Mass + Spring	0.291	0.283	0.291	**0.278**	0.291	0.241
Spring Pendulum	0.403	0.392	0.403	**0.390**	0.403	0.332

Using the ODE data set, we observe that DaD_sub$^+$ outperforms all iterative approaches on all the data sets considered and is close to the best one (direct approach). Indeed, the direct approach performs better than all other methods and gives us a measure of comparison for methods with a specific objective, i.e., to predict at a fixed step. Figure 6.5 displays multi-step losses of the best DaD_sub$^+$ model trained at different horizon h_k. The heatmap shows the benefit of using incremental learning. In fact, using corrections improves the model's ability to learn at long-term horizons.

Regarding interpretability, when the data is generated by an ODE and observed at discrete times, the definition of the set of true coefficients T is challenging. We nevertheless provide results and discussion of this point in the appendix.

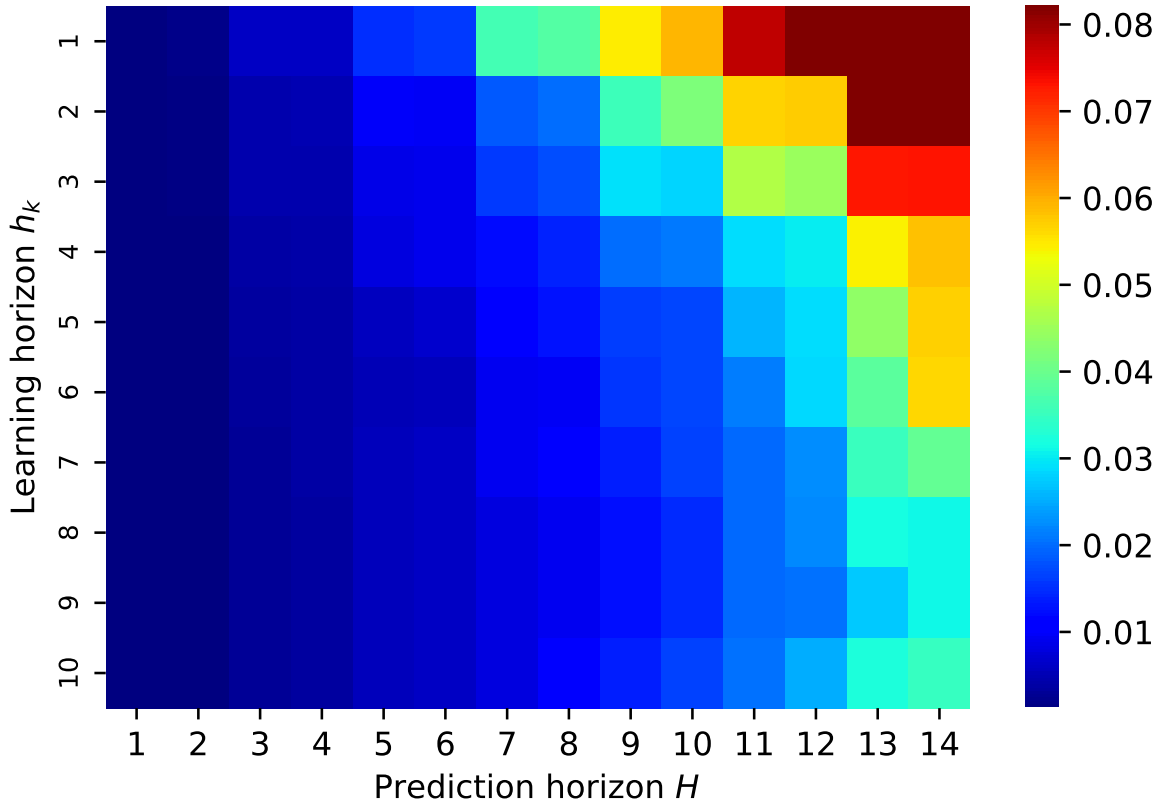


Figure 6.5: Evolution of the best prediction loss for models trained for a horizon h_k through DaD_sub^+ iterations against horizon H (averaged over 50 runs).

6.4.4.5 Real world data-set

The visualization of the prediction loss evolution in Figure 6.6 provides a compelling illustration of the effectiveness of our approach and reinforces the credibility of our findings. Here, the interpretability and analysis of the coefficients are limited as we do not know the true coefficient set T, or even if it exists. Nevertheless, our framework shows that DaD_sub^+ is the sparsest model and that the variables are consistent with domain knowledge.

6.4.4.6 Limitations of our approach

The interpretability analysis of time series data can be challenging due to the absence of a unique solution and the ability to make long-term accurate forecasts with coefficients that differ from the true ones. In such cases, the sparsity criterion gives a practical measure for model interpretability. Moreover, including extended non-linear expansion in the model increases computational complexity and multicollinearity, which is another limitation to consider.

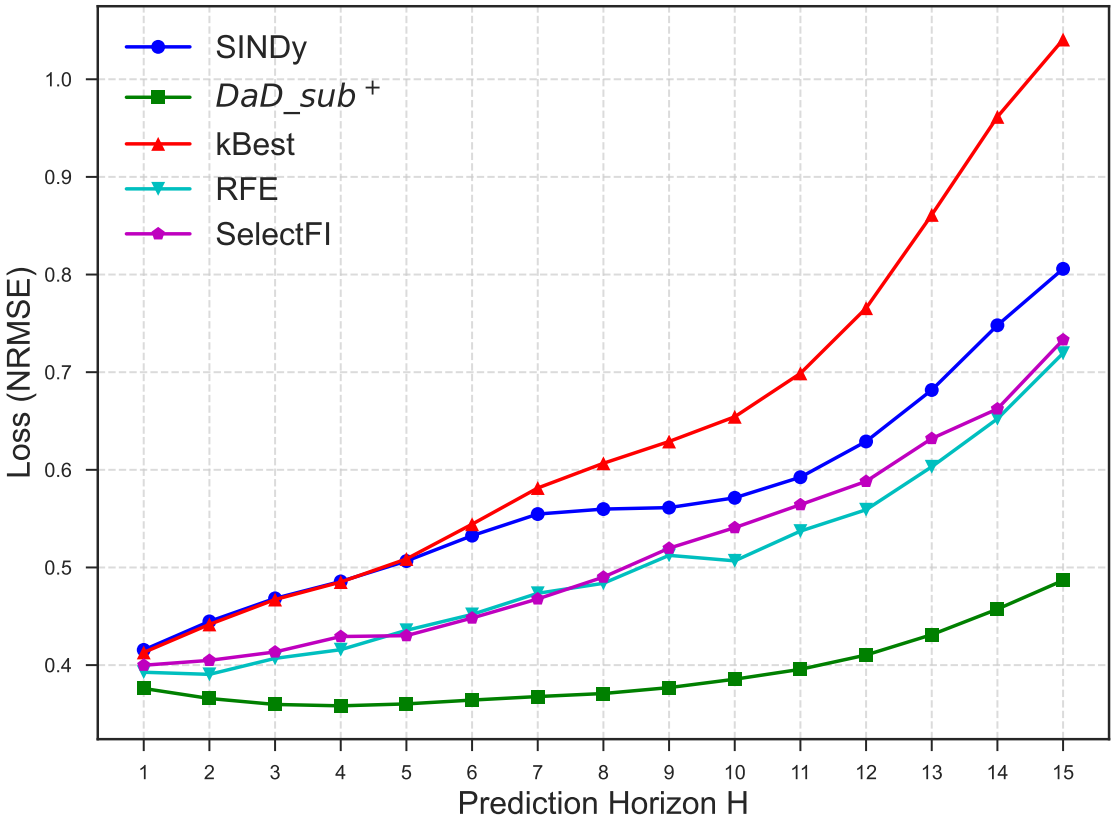


Figure 6.6: Prediction loss comparison against horizon for wind power forecasting

6.5 Conclusion and future works

In this work, we address the problem of multi-step ahead prediction of multivariate time series by learning an interpretable and coherent model. We develop a framework to improve a linear predictor by efficiently using training data using the DAgger approach. We present methodological, theoretical, and experimental contributions.
Future works should adapt this approach to a broader range of models, allowing the direct method to be challenged without compromising interpretability. Moreover, further analyses will be made to evaluate this framework's ability to identify the parameters of the underlying dynamics in the presence of multicollinearity.

CHAPTER 7

Conclusion and Perspectives

Contents

7.1 Conclusion

In this thesis, we aimed to develop algorithms and methods for analyzing time series in a simple and comprehensible way. After describing the field of XAI and clarifying the definition of interpretability, we focused on developing these models to improve existing transparent approaches by increasing their performance or building new interpretable methods.

From data measured over time, such as sensors on a machine or the monitoring of variables, the aim is to extract patterns and knowledge to make effective decisions. Given business constraints, the models developed must be not only efficient but also transparent in order to reveal the decision-making process clearly by describing the mechanisms relating the input variables to the output decision. This increases acceptance and management of these tools.

Two main problems were addressed in this thesis, leading to two contributions.

Root Cause analysis

Many complex models allow a labeled data set to learn and predict the occurrence of events, such as failures from a time series data set containing a large number of variables. Even though high accuracy can be achieved, some errors and missed events can potentially lead to huge losses due to the lack of decision-making and model understanding. The first works tackled this challenge by proposing an interpretable approach to uncover the root cause of failures.

Our approach aims to build simple rules using an association rule mining framework. The first step involves transforming and discretizing the data set to create a new database for the inference. The second step involves incorporating causality as association rules mining, as its name indicates, only deals with associations. This is done by taking advantage of an epidemiological approach called the case-crossover design. The combination of these two approaches provides causal and interpretable rules for understanding the causes of the failure.

Then, two predictive algorithms were constructed based on the causal rules, allowing forecasting of the occurrence of the failures. The first algorithms select rules based on several criteria and then aggregate the rule's decisions to make a global prediction. The second algorithm takes advantage of the first and improves it by adding anti-rules, which are the rules that predict a normal situation. The aggregation for both rules allows for a powerful predictive algorithm.

This approach was tested on a real-world data set where a phenomenon called flooding happens briefly in time and induces a failure. The objective was to identify the causes of the problem and provide operators with simple and interpretable information. The approach and the predictive algorithms were applied to this data set and showed strong performance, which experts in the domain validated.

Dynamic modeling

After providing the tools for analyzing the causes of specific phenomena from potentially high-dimensional multivariate time series data, we approached the problem from a global perspective, learning the underlying dynamics generating the time series. The aim was to extract an equation from the observations, enabling us to determine the relationships between the explanatory variables and an output variable. This equation yields an interpretable model that allows a better understanding of the phenomenon at play and the underlying systems.

We have developed an approach for learning these dynamics from interpretable methods such as vector autoregressive models. The input dimensions were increased by introducing non-linearities in the input variables to account for the real data's non-linear nature. Two main problems associated with this approach arose: the high-dimensional nature of the study and the lack of performance of regression methods, particularly in a multi-step learning setting where error propagation occurs.

To address the first problem, applying penalized regression methods that encourage sparsity enables the selection of a restricted set of variables, considerably reducing the number of variables and selecting only the most significant ones. To overcome the second problem, we have combined these penalized regression methods with imitative learning methods to correct error propagation. The proposed iterative algorithm learns a set of candidate models, learning the dynamics and successively correcting the trajectories. After learning the dynamics, the model selection is then made by cross-validation.

This approach was tested on synthetic time series data from ODEs to recover the equations of this discretized ODE. Once validated using forecasting performance metrics and interpretability analyses, we applied our model to a wind data set.

7.2 Perspectives

This thesis lays the groundwork for further research. We discuss the challenges and opportunities for further progress raised by the two approaches developed.

Causal rules algorithm

Improving time series discretization and labelization An implicit assumption of the developed approach is that the discretized items contain the causal information and potentially match the causal scale (Gong et al., 2015; Shojaie & Fox, 2022). Consequently, one area for improvement in the association rule algorithm lies in discretization. This necessary step, involving a symbolic representation of the time series, creates a loss of information. This can be reduced using algorithms such as SAX (Symbolic Aggregate Approximation) (Lin et al., 2007), which involves two main steps: first, it automatically reduces the dimensionality of time series by aggregating data points. Secondly, it converts the reduced data into a symbolic representation using predefined symbols. In addition, further evaluations on synthetic datasets where the ground truth is known are essential to determining precisely how effective the algorithm is in mitigating information loss.

Another limitation is related to data labeling. As the root cause of the failures is unknown, labels were assigned based on information that could come from different places in the causal chain. This introduces uncertainty in the identification of the exact cause. In addition, the labeling process involved differentiating between normal and abnormal operating conditions, including their durations. While expert knowledge facilitated this labeling, further research could explore the potential for automating this process.

Granger causality and Interventions Granger's causal analysis relies on specific statistical assumptions such as causal sufficiency and stationarity. Spurious correlations may be identified instead of true causal relationships if these assumptions are not verified. Several methodologies

have been proposed to address these limitations in the context of vector autoregressive (VAR) models (Shojaie & Fox, 2022). Future research efforts should focus on critically evaluating our association rule algorithm's assumptions related to Granger causality and the case-crossover design to ensure accurate identification of causal effects. However, it is important to recognize that assumptions such as causal sufficiency cannot be tested and, therefore, necessitate the expertise of a domain expert.

On the other hand, an open question is how to prevent the occurrence of failure and what should be done to avoid future occurrences. To achieve this, we need to identify the variables that can be manipulated in the causal graph and intervene to defuse potential failures. The causality graph enables us to make deductions and develop targeted interventions to identify the root cause and the variables that prevent such incidents. Under certain assumptions (Assaad et al., 2023; Pearl, 2000), we can assess changes in the causal mechanism by intervening on the causal graph. The information obtained can then be used to develop actionable intervention strategies.

Allowing for non-observed variables Numerous sensors can be placed on a system to capture its evolution, but additional unobserved variables can act on the system. Adding these variables to the analysis could provide additional information for the overall understanding of the system (Strobl & Lasko, 2023).

Dynamic discovery

Theoretical limitation Some important aspects still need to be addressed (Camps-Valls et al., 2023). First, we need to determine whether the system of equations is identifiable, i.e., whether the structure and the parameters can be recovered from the data. Second, current selection methods, which often favor models with fewer parameters (sparsity), may not be sufficient. A more comprehensive evaluation approach is necessary to identify the best model. Additionally, our current analysis assumes all variables are observed. However, future studies could explore the potential influence of latent variables (unobserved factors) on the system.

In-depth study on the multicollinearity problem An important limitation to discovering underlying dynamical systems is the presence of multiple variables and, consequently, multicollinearity in the dataset. This can prevent accurate parameter estimation and identification of the true dynamics of the system. Regularization techniques like Lasso allow for reducing the number of variables, but these methods choose variables at random from a set of correlated groups. To ensure the identification of the true system, consistency conditions such as the irrepresentable conditions (Zhao & Yu, 2006) should be integrated into our approach. Additionally, alternative feature selection approaches require careful consideration to avoid discarding important variables.

Alternative to Lasso Our current approach uses the Lasso algorithm, which can be quite slow when dealing with high-dimensional data. This presents a challenge for the iterative nature of our dynamic discovery algorithm. Therefore, exploring alternative approaches for faster computation is an area of research that we need to focus on in the future. One possible solution could be using the sparse relaxed regularized regression (SR3) problem, which has

been proposed in recent studies (Champion et al., 2020; Rudy et al., 2017; Zheng et al., 2018). Researching and experimenting with this approach could benefit our algorithm.

Appendices

APPENDIX A

Appendix for chapter 3

Conditional logistic regression

Let $i \in \{1, \dots, N\}$ be the subject that is studied, $\mathbf{x}_{i,1}$ a realization of $\mathbf{X}_1$ the random vector of exposure for the case, $\mathbf{x}_{i,0}$ of $\mathbf{X}_0$ the exposure vector for the control, and $Y_k \in \{0, 1\}$ be the associated outcome variable for $k \in \{0, 1\}$. Both $\mathbf{X}_1$ and $\mathbf{X}_0$ take discrete values in $\mathbb{R}^p$. Let us prove that the conditional likelihood is written:

$$l(\beta) = \prod_{i=1}^{N} \frac{\exp(\beta^T \mathbf{x}_{i,1})}{\exp(\beta^T \mathbf{x}_{i,0}) + \exp(\beta^T \mathbf{x}_{i,1})} = \prod_{i=1}^{N} \frac{1}{1 + \exp(\beta^T (\mathbf{x}_{i,0} - \mathbf{x}_{i,1}))} \tag{A.1}$$

Let $\mathbf{x}_{i,0}, \mathbf{x}_{i,1} \in \mathbb{R}^p$, the logistic function is defined as

$$P(Y_1 = 1 | \mathbf{X}_1 = \mathbf{x}_{i,1}) = \frac{\exp(\lambda_i + \beta^T \mathbf{x}_{i,1})}{1 + \exp(\lambda_i + \beta^T \mathbf{x}_{i,1})} \tag{A.2}$$

$$P(Y_0 = 0 | \mathbf{X}_0 = \mathbf{x}_{i,0}) = 1 - P(Y_0 = 1 | \mathbf{X} = \mathbf{x}_{i,0}) = \frac{1}{1 + \exp(\lambda_i + \beta^T \mathbf{x}_{i,0})} \tag{A.3}$$

where $\beta \in \mathbb{R}^p$ is the regression vector and λ_i is the intercept.

Let us consider $(\mathbf{X}_0, \mathbf{X}_1, Y_0, Y_1)$ and assuming that $(\mathbf{X}_0, Y_0)$ and $(\mathbf{X}_1, Y_1)$ are conditionally independent, we compute the probability of the event $(\mathbf{X}_0 = \mathbf{x}_{i,0}, \mathbf{X}_1 = \mathbf{x}_{i,1})$ conditioned on $(Y_0 = 0, Y_1 = 1)$ using Bayes formula:

$$\begin{aligned} l(\beta) &= P\left(\mathbf{X}_0 = \mathbf{x}_{i,0}, \mathbf{X}_1 = \mathbf{x}_{i,1} | Y_0 = 0, Y_1 = 1\right) \\ &= \frac{P\left(\mathbf{X}_0 = \mathbf{x}_{i,0}, \mathbf{X}_1 = \mathbf{x}_{i,1}, Y_0 = 0, Y_1 = 1\right)}{P\left(Y_0 = 0, Y_1 = 1\right)} \\ &= \frac{P\left(Y_0 = 0, Y_1 = 1 | \mathbf{X}_0 = \mathbf{x}_{i,0}, \mathbf{X}_1 = \mathbf{x}_{i,1}\right) P\left(\mathbf{X}_0 = \mathbf{x}_{i,0}, \mathbf{X}_1 = \mathbf{x}_{i,1}\right)}{P\left(Y_0 = 0, Y_1 = 1\right)} \end{aligned} \tag{A.4}$$

By independence, we could rewrite it

$$l(\beta) = \frac{P\left(Y_0 = 0 | \mathbf{X}_0 = \mathbf{x}_{i,0}\right) P\left(Y_1 = 1 | \mathbf{X}_1 = \mathbf{x}_{i,1}\right) P\left(\mathbf{X}_0 = \mathbf{x}_{i,0}\right) P\left(\mathbf{X}_1 = \mathbf{x}_{i,1}\right)}{P\left(Y_0 = 0\right) P\left(Y_1 = 1\right)} \tag{A.5}$$

To estimate β, we need to develop equation A.5 to do Maximum Likelihood Estimation (MLE). In this equation, only $P(\mathbf{X}_k = \mathbf{x}_{i,k})$ and $P(Y_k = k)$ for $k \in \{0, 1\}$ are unknown. $P(\mathbf{X}_0 = \mathbf{x}_{i,0}) P(\mathbf{X}_1 = \mathbf{x}_{i,1})$ can be considered as a constant as it does not depend on β. Let us expand $P(Y_k = k)$ using the law of total probability:

$$P(Y_k = k) = \sum_{\mathbf{x}'} P(Y_k = k | \mathbf{X}_k = \mathbf{x}') P(\mathbf{X}_k = \mathbf{x}') \tag{A.6}$$

Here, we see the limits of the logistic regression model because we need to know the distribution of $\mathbf{X}_k$ to be able to know the probability (A.6) and apply MLE.

In conditional logistic regression, an alternative is proposed when the distribution of $\mathbf{X}_k$ is unknown. Instead of calculating the probability as in equation A.5, we calculate the probability of the event $(\mathbf{X}_0 = \mathbf{x}_{i,0}, \mathbf{X}_1 = \mathbf{x}_{i,1})$ among $\mathrm{S} = \{(\mathbf{x}_{i,0}, \mathbf{x}_{i,1}), (\mathbf{x}_{i,1}, \mathbf{x}_{i,0})\}$ conditioned on $(Y_0 = 0, Y_1 = 1)$. This can written as:

$$\begin{aligned}
l(\beta) &= \frac{P(\mathbf{X}_1 = \mathbf{x}_{i,1}, \mathbf{X}_0 = \mathbf{x}_{i,0} | Y_0 = 0, Y_1 = 1)}{P(\mathbf{X}_1 = \mathbf{x}_{i,1}, \mathbf{X}_0 = \mathbf{x}_{i,0} | Y_0 = 0, Y_1 = 1) + P(\mathbf{X}_1 = \mathbf{x}_{i,0}, \mathbf{X}_0 = \mathbf{x}_{i,1} | Y_0 = 0, Y_1 = 1)} \\
&= \frac{\frac{P(Y_0=0,Y_1=1|\mathbf{X}_0=\mathbf{x}_{i,0},\mathbf{X}_1=\mathbf{x}_{i,1})P(\mathbf{X}_0=\mathbf{x}_{i,0})P(\mathbf{X}_1=\mathbf{x}_{i,1})}{P(Y_0=0,Y_1=1)}}{\frac{P(Y_0=0,Y_1=1|\mathbf{X}_0=\mathbf{x}_{i,0},\mathbf{X}_1=\mathbf{x}_{i,1})P(\mathbf{X}_0=\mathbf{x}_{i,0})P(\mathbf{X}_1=\mathbf{x}_{i,1})}{P(Y_0=0,Y_1=1)} + \frac{P(Y_0=0,Y_1=1|\mathbf{X}_1=\mathbf{x}_{i,0},\mathbf{X}_0=\mathbf{x}_{i,1})P(\mathbf{X}_0=\mathbf{x}_{i,1})P(\mathbf{X}_1=\mathbf{x}_{i,0})}{P(Y_0=0,Y_1=1)}} \\
&= \frac{P(Y_1 = 1 | \mathbf{X}_1 = \mathbf{x}_{i,1}) \cdot P(Y_0 = 0 \mid \mathbf{X}_0 = \mathbf{x}_{i,0})}{P(Y_1 = 1 | \mathbf{X}_1 = \mathbf{x}_{i,1}) \cdot P(Y_0 = 0 \mid \mathbf{X}_0 = \mathbf{x}_{i,0}) + P(Y_0 = 0 | \mathbf{X}_1 = \mathbf{x}_{i,0}) \cdot P(Y_1 = 1 \mid \mathbf{X}_0 = \mathbf{x}_{i,1})}
\end{aligned} \tag{A.7}$$

Using equation A.2 and A.3, we can write the conditional likelihood function

$$\begin{aligned}
l(\beta) &= \frac{P(Y_1 = 1 | \mathbf{X}_1 = \mathbf{x}_{i,1}) \cdot P(Y_0 = 0 \mid \mathbf{X}_0 = \mathbf{x}_{i,0})}{P(Y_1 = 1 | \mathbf{X}_1 = \mathbf{x}_{i,1}) \cdot P(Y_0 = 0 \mid \mathbf{X}_0 = \mathbf{x}_{i,0}) + P(Y_0 = 0 | \mathbf{X}_1 = \mathbf{x}_{i,0}) \cdot P(Y_1 = 1 \mid \mathbf{X}_0 = \mathbf{x}_{i,1})} \\
&= \frac{\frac{\exp(\lambda_i + \beta^T \mathbf{x}_{i,1})}{1+\exp(\lambda_i + \beta^T \mathbf{x}_{i,1})} \cdot \frac{1}{1+\exp(\lambda_i + \beta^T \mathbf{x}_{i,0})}}{\frac{\exp(\lambda_i + \beta^T \mathbf{x}_{i,1})}{1+\exp(\lambda_i + \beta^T \mathbf{x}_{i,1})} \cdot \frac{1}{1+\exp(\lambda_i + \beta^T \mathbf{x}_{i,0})} + \frac{1}{1+\exp(\lambda_i + \beta^T \mathbf{x}_{i,1})} \cdot \frac{\exp(\lambda_i + \beta^T \mathbf{x}_{i,0})}{1+\exp(\lambda_i + \beta^T \mathbf{x}_{i,0})}} \\
&= \frac{\exp\left(\lambda_i + \beta^\top \mathbf{x}_{i,1}\right)}{\exp(\lambda_i + \beta^\top \mathbf{x}_{i,1}) + \exp(\lambda_i + \beta^\top \mathbf{x}_{i,0})}
\end{aligned}$$

By dividing by the numerator, this leads to

$$l(\beta) = \frac{1}{1 + \exp(\beta^\top (\mathbf{x}_{i,0} - \mathbf{x}_{i,1}))} \tag{A.8}$$

This is the logistic regression for subject i. The case crossover is done here with N observations with $i \in \{1, \ldots, N\}$. Hence, the conditional likelihood function is written

$$L(\beta) = \prod_{i=1}^{N} \frac{1}{1 + \exp(\beta^\top (\mathbf{x}_{i,0} - \mathbf{x}_{i,1}))} \tag{A.9}$$

APPENDIX B

Appendix for chapter 6

This appendix is organized as follows. Section B.1 presents additional information on the data-sets, and Section B.2 gives additional results from the experiments.

B.1 Data-set Details

In this section, we describe the non-linear equations used to generate the data-sets used in Section 5, and we plot multivariate time series for each of them. All physical quantities are expressed in standard units (and $g = 9.81 m.s^{-2}$ is the gravitational acceleration constant).

B.1.1 Cartpole

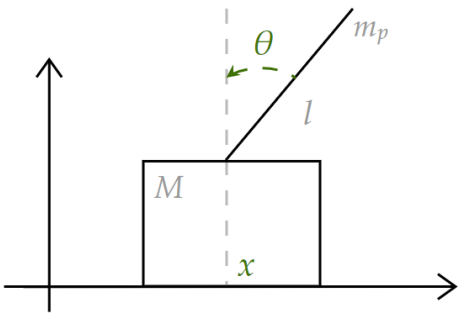


Figure S1: Cartpole

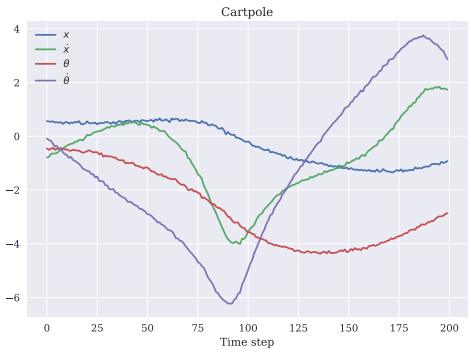


Figure S2: Variables: $x, \dot{x}, \theta, \dot{\theta}$

The first system that we study is the cartpole. Let $l = 1$ and $m_p = m_c = 1$ be respectively the length and the masses. The data-sets are generated using the following equations:

$$\ddot{\theta} = \frac{g \sin\theta + \cos\theta \left(\frac{-F - m_p l \dot{\theta}^2 \sin\theta}{m_c + m_p} \right)}{l \left(\frac{4}{3} - \frac{m_p \cos^2\theta}{m_c + m_p} \right)}$$

$$\ddot{x} = \frac{F + m_p l \left(\dot{\theta}^2 \sin\theta - \ddot{\theta} \cos\theta \right)}{m_c + m_p}.$$

B.1.2 Pendulum

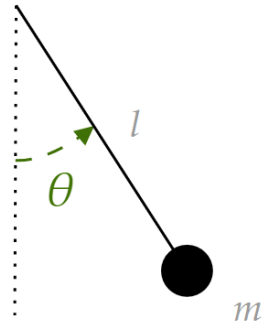


Figure S3: Pendulum

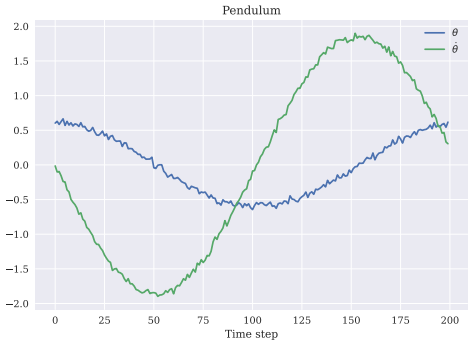


Figure S4: Variables: $\theta, \dot{\theta}$

The second system is a single pendulum. Let $l = 1$, $m = 1$, and $c = 0.005$ be, respectively, the pendulum's length, mass, and dampening factor. The data-sets are generated using the following equations:

$$\ddot{\theta} = \frac{c}{m}\dot{\theta} - \frac{g}{l}\sin(\theta).$$

B.1.3 Double Pendulum

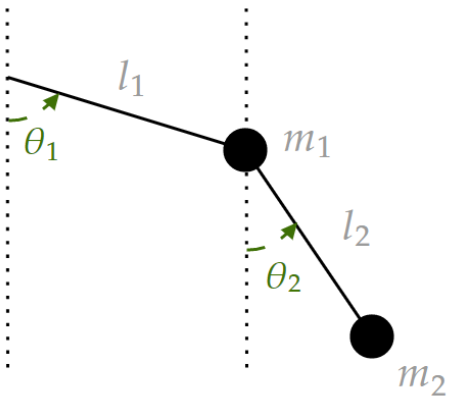


Figure S5: Double Pendulum

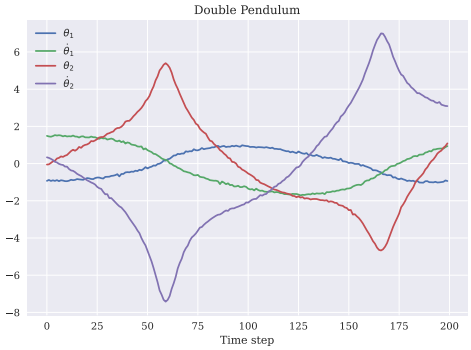


Figure S6: Variables: $\theta_1, \dot{\theta}_1, \theta_2, \dot{\theta}_2$

Here, we describe the equations of a double pendulum. Let $m_1 = m_2 = 1$, $l_1 = l_2 = 1$, $c = 0.005$ and $\Delta\theta = \theta_1 - \theta_2$. We have the following equations:

$$\ddot{\theta}_1 = \frac{m_2 l_1 \dot{\theta}_1^2 \sin(2\Delta\theta) + 2m_2 l_2 \dot{\theta}_2^2 \sin\Delta\theta + 2gm_2 \cos\theta_2 \sin\Delta\theta + 2gm_1 \sin\theta_1}{-2l_1\left(m_1 + m_2 \sin^2\Delta\theta\right)},$$

$$\ddot{\theta}_2 = \frac{m_2 l_2 \dot{\theta}_2^2 \sin(2\Delta\theta) + 2\left(m_1 + m_2\right) l_1 \dot{\theta}_1^2 \sin\Delta\theta + 2g\left(m_1 + m_2\right)\cos\theta_1 \sin\Delta\theta}{2l_2\left(m_1 + m_2 \sin^2\Delta\theta\right)}.$$

B.1.4 Spring Pendulum

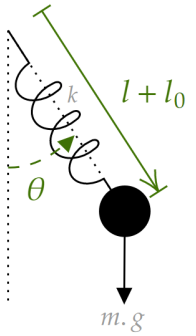


Figure S7: Spring Pendulum

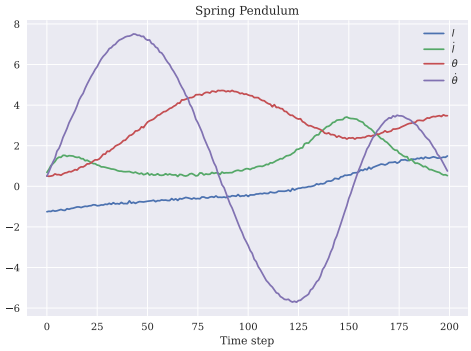


Figure S8: Variables: $l, \dot{l}, \theta, \dot{\theta}$

The system we study here is a spring pendulum. Let $l = 2$, $k = 40$ and $m = 3$. The data-sets are generated using the following equations:

$$\ddot{\theta} + \frac{1}{l}\left(g \sin\theta + 2\dot{\theta}\dot{l}\right) = 0$$
$$\ddot{l} + \frac{1}{m}\left(k(l - l_0) - ml\dot{\theta}^2 + mg\cos\theta\right) = 0$$

B.1.5 System Mass Spring Pendulum

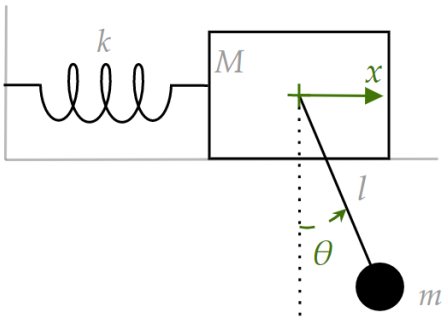


Figure S9: Mass Spring Pendulum

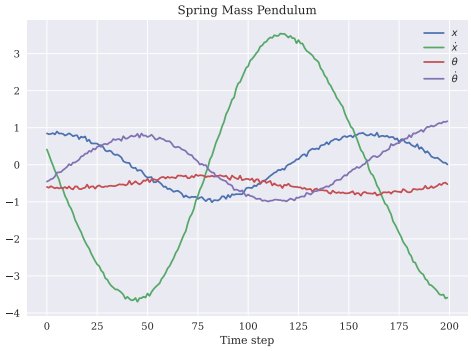


Figure S10: Variables: $x, \dot{x}, \theta, \dot{\theta}$

This system comprises a mass, a spring, and a pendulum. Let $\mu = \frac{m}{m+M} = 0.4$ and $\varepsilon = 0.3$. The data-sets are generated using the following equations:

$$\ddot{\theta} + \frac{1}{1 - \mu\cos^2\theta}\left(\sin\theta + \mu\dot{\theta}^2\cos\theta\sin\theta - \frac{1}{\varepsilon}x\cos\theta\right) = 0$$
$$\ddot{x} + \frac{1}{1 - \mu\cos^2\theta}\left(\frac{1}{\varepsilon^2}x - \frac{\mu}{\varepsilon}\left(\dot{\theta}^2\sin\theta + \cos\theta\sin\theta\right)\right) = 0$$

B.1.6 Discrete Lotka-Volterra

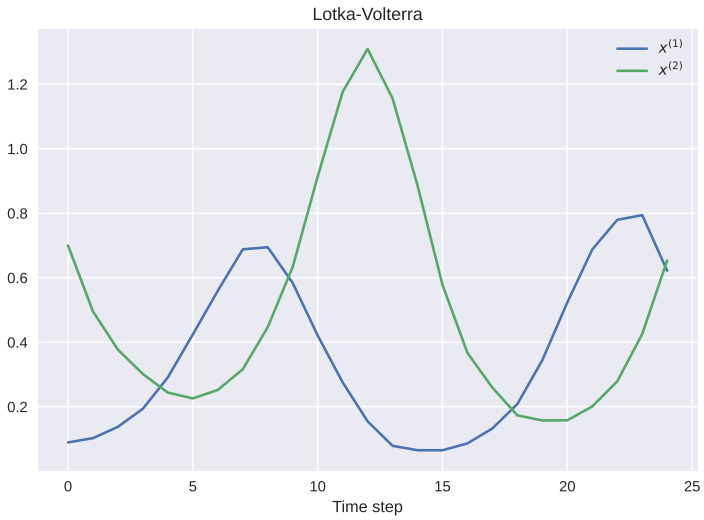


Figure S11: Lotka-Volterra

This system models the interaction between two species: a predator and its prey. Let $\delta = 1$ and $\varepsilon \sim \mathcal{N}(0, 1)$. The data-sets are generated using the following equations:

$$x_{t+1}^{(1)} = (1+\delta)x_t^{(1)} - \delta(x_t^{(1)})^2 - \delta x_t^{(1)} x_t^{(2)} + \varepsilon_t^1$$
$$x_{t+1}^{(2)} = (1-0.5\delta)x_t^{(2)} - 0.1\delta(x_t^{(2)})^2 + 2.1\delta x_{t-1}^{(1)} x_{t-1}^{(2)} + \varepsilon_t^2$$

B.1.7 Discrete SIR

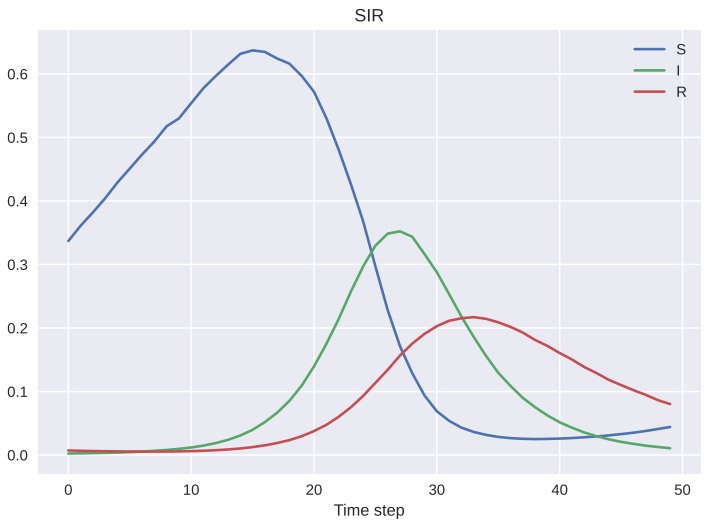


Figure S12: SIR

The SIR model is a set of differential equations that describe the dynamics of an infectious disease in a population. The population is divided into three compartments: Susceptible (S), Infected (I), and Recovered (R). Let $\delta = 1$, $r = 0.1$, $\mu = 0.1$, $\alpha = 0$, $\gamma = 0.1$ and $\varepsilon \sim \mathcal{N}(0,\, 1)$. The data-sets are generated using the following equations:

$$
\begin{aligned}
S_{t+1} &= S_t + \delta r S_t - r\delta S_t^2 - \ \delta S_t I_{t-1}/(1 + \alpha S_t) + \varepsilon_t^1 \\
I_{t+1} &= I_t - (\mu + \gamma)\delta I_t + \delta S_t I_{t-1}/(1 + \alpha I_t) + \varepsilon_t^2 \\
R_{t+1} &= R_t - \mu\delta R_t + \delta\gamma I_t + \varepsilon_t^3
\end{aligned}
$$

B.2 Additional Experimental Results

This section presents a detailed presentation of additional experimental results. First, we conduct an analysis of the influence of the threshold α on the interpretability loss. Finally, we provide additional findings pertaining to the impact of data-set size and noise level on our discrete time series experiments.

B.2.1 Choice of the correlation threshold α

To take into account the collinearity in the data, we define in section 5.1 of the main article the sets $\mathrm{T}^+ = \{i | j \in \mathrm{T}, |\mathrm{corr}[i, j]| \geq \alpha\}$ and $\mathrm{P}^+ = \{i | j \in \mathrm{P}, |\mathrm{corr}[i, j]| \geq \alpha\}$ which are, respectively, the variables correlated more than a threshold $\alpha \in [0, 1]$ with the true variables and the predicted variables. In the following, we show the effect of this parameter α on interpretability metrics against noise level and data-set sizes. We add two metrics $\frac{|\mathrm{T}^+|}{p_\Phi}$ and $\frac{|\mathrm{P}^+|}{p_\Phi}$ that measure the ratio of variables that are considered to be correlated with the true and predicted ones, respectively.

Noise	Metric	$\alpha = 0.8$	$\alpha = 0.9$	$\alpha = 0.95$	$\alpha = 0.98$
0.01	$\overline{\text{F2-score}}$	0.972±0.0	0.931±0.001	0.874±0.006	0.729±0.002
	$\overline{\text{Precision}}$	0.886±0.001	0.739±0.002	0.591±0.015	0.464±0.005
	$\overline{\text{Recall}}$	1.0±0.0	1.0±0.0	1.0±0.0	0.833±0.0
	$\overline{\text{FP}}$	1.0±0.0	2.034±0.012	2.929±0.081	4.017±0.039
	$\frac{\lvert\text{T}^+\rvert}{p_\Phi}$	0.603±0.0	0.427±0.0	0.338±0.0	0.179±0.001
	$\frac{\lvert\text{P}^+\rvert}{p_\Phi}$	0.866±0.002	0.773±0.006	0.573±0.001	0.383±0.003
0.03	$\overline{\text{F2-score}}$	0.973±0.001	0.896±0.008	0.868±0.01	0.646±0.017
	$\overline{\text{Precision}}$	0.888±0.003	0.653±0.026	0.587±0.031	0.404±0.029
	$\overline{\text{Recall}}$	1.0±0.0	1.0±0.0	1.0±0.0	0.793±0.047
	$\overline{\text{FP}}$	0.986±0.016	2.636±0.258	3.02±0.314	4.398±0.378
	$\frac{\lvert\text{T}^+\rvert}{p_\Phi}$	0.603±0.0	0.426±0.0	0.337±0.001	0.169±0.001
	$\frac{\lvert\text{P}^+\rvert}{p_\Phi}$	0.867±0.004	0.735±0.009	0.551±0.007	0.384±0.029
0.07	$\overline{\text{F2-score}}$	0.976±0.001	0.894±0.014	0.835±0.016	0.604±0.041
	$\overline{\text{Precision}}$	0.9±0.002	0.698±0.014	0.546±0.008	0.424±0.01
	$\overline{\text{Recall}}$	1.0±0.0	0.975±0.021	0.972±0.024	0.72±0.068
	$\overline{\text{FP}}$	0.81±0.052	2.15±0.139	2.769±0.118	3.711±0.193
	$\frac{\lvert\text{T}^+\rvert}{p_\Phi}$	0.592±0.003	0.424±0.001	0.304±0.001	0.135±0.002
	$\frac{\lvert\text{P}^+\rvert}{p_\Phi}$	0.817±0.009	0.681±0.021	0.479±0.021	0.282±0.024
0.1	$\overline{\text{F2-score}}$	0.976±0.003	0.89±0.022	0.807±0.024	0.619±0.042
	$\overline{\text{Precision}}$	0.9±0.009	0.719±0.03	0.531±0.022	0.454±0.029
	$\overline{\text{Recall}}$	1.0±0.0	0.961±0.017	0.931±0.012	0.722±0.055
	$\overline{\text{FP}}$	0.776±0.031	2.034±0.085	2.827±0.091	3.398±0.11
	$\frac{\lvert\text{T}^+\rvert}{p_\Phi}$	0.592±0.0	0.409±0.001	0.246±0.003	0.119±0.001
	$\frac{\lvert\text{P}^+\rvert}{p_\Phi}$	0.79±0.012	0.632±0.011	0.39±0.009	0.235±0.005

Table B.1: Study of the correlation parameter α against noise levels: Comparison of different values of α over noise levels on interpretability metrics, F2-score, Precision, Recall, FP, and the ratio of the correlated variable with true and predicted ones. The mean of metric X, denoted as $\overline{X}$, and standard deviation are computed over 50 runs.

Data-set Size	Metric	$\alpha = 0.8$	$\alpha = 0.9$	$\alpha = 0.95$	$\alpha = 0.98$
2	$\overline{\text{F2-score}}$	0.973±0.002	0.894±0.026	0.839±0.046	0.624±0.074
	$\overline{\text{Precision}}$	0.89±0.006	0.699±0.032	0.567±0.047	0.437±0.023
	$\overline{\text{Recall}}$	1.0±0.0	0.975±0.029	0.967±0.04	0.723±0.084
	$\overline{\text{FP}}$	0.885±0.117	2.143±0.203	2.781±0.115	3.747±0.298
	$\frac{\lvert\text{T}^+\rvert}{p_\Phi}$	0.598±0.005	0.421±0.009	0.306±0.042	0.151±0.028
	$\frac{\lvert\text{P}^+\rvert}{p_\Phi}$	0.834±0.034	0.7±0.06	0.493±0.082	0.306±0.074
5	$\overline{\text{F2-score}}$	0.975±0.002	0.903±0.02	0.847±0.033	0.652±0.056
	$\overline{\text{Precision}}$	0.896±0.009	0.703±0.031	0.566±0.038	0.435±0.026
	$\overline{\text{Recall}}$	1.0±0.0	0.983±0.02	0.975±0.032	0.771±0.056
	$\overline{\text{FP}}$	0.883±0.131	2.214±0.231	2.901±0.057	3.921±0.41
	$\frac{\lvert\text{T}^+\rvert}{p_\Phi}$	0.597±0.007	0.422±0.008	0.306±0.043	0.15±0.028
	$\frac{\lvert\text{P}^+\rvert}{p_\Phi}$	0.836±0.036	0.706±0.058	0.5±0.08	0.324±0.075
15	$\overline{\text{F2-score}}$	0.975±0.003	0.91±0.018	0.852±0.017	0.674±0.039
	$\overline{\text{Precision}}$	0.895±0.01	0.705±0.058	0.559±0.012	0.438±0.049
	$\overline{\text{Recall}}$	1.0±0.0	0.994±0.01	0.985±0.028	0.807±0.031
	$\overline{\text{FP}}$	0.911±0.11	2.283±0.449	2.977±0.282	3.974±0.618
	$\frac{\lvert\text{T}^+\rvert}{p_\Phi}$	0.596±0.008	0.422±0.008	0.306±0.045	0.15±0.029
	$\frac{\lvert\text{P}^+\rvert}{p_\Phi}$	0.834±0.045	0.709±0.069	0.502±0.088	0.334±0.078

Table B.2: Study of the correlation parameter α against data-set sizes: Comparison of different values of α over data-set sizes on interpretability metrics, F2-score, Precision, Recall, FP and ratio of correlated variable with true and predicted ones. The mean of metric X, denoted as $\overline{X}$, and standard deviation are computed over 50 runs.

For a comprehensive evaluation of interpretability, selecting an α parameter that identifies a reasonable quantity of variables is essential. Selecting all variables would result in optimal scores for the F2-score, Precision, Recall, Sparseness, and FP metrics. However, this approach may not provide a fair assessment. Upon examination of Tables B.1 and B.2, we have chosen an α value of 0.9. This value appears to achieve a trade-off between the number of variables correlated with the true variables and those correlated with the predictions.

B.2.2 Effect of Noise on Interpretability and Prediction

In this section, we study the effect of added noise in the system on interpretability and prediction in the Lotka-Volterra and SIR data-sets. The data presented in Tables B.3 and B.4 provide a comparative analysis of the performance of various methods under different noise levels. The tables demonstrate that when compared to other methods, DaD_sub$^+$ consistently

exhibits superior performance across all noise levels. In fact, DaD_sub$^+$ consistently secures a position within the top two ranks in all scenarios. This consistent high-ranking performance of DaD_sub$^+$, irrespective of the noise level, underscores its robustness and effectiveness.

Lotka-Volterra

Noise	Metric	SINDy	DaD_sub$^+$	kBest	RFE	SelectFI
0.01	$\overline{\text{F2-score}}$	0.931±0.003	**0.931**±0.001	0.837±0.018	0.794±0.022	0.703±0.061
	$\overline{\text{Precision}}$	0.74±0.008	**0.739**±0.003	0.629±0.015	0.657±0.047	0.75±0.0
	$\overline{\text{Recall}}$	**1.0**±0.0	**1.0**±0.0	0.928±0.04	0.839±0.056	0.57±0.116
	$\overline{\text{FP}}$	2.262±0.042	2.034±0.012	10.13±0.544	3.063±0.656	**1.503**±0.006
	$\overline{\text{Sparsity}}$	2.721±0.089	2.441±0.026	6.92±0.439	2.691±0.371	**1.539**±0.064
	$\overline{\text{NRMSE}}$	**3.6e-04**±6e-05	3.7e-04±3e-05	/	0.431±0.001	0.431±0.001
0.03	$\overline{\text{F2-score}}$	0.882±0.008	**0.894**±0.007	0.83±0.009	0.816±0.021	0.711±0.07
	$\overline{\text{Precision}}$	0.61±0.024	0.653±±0.026	0.632±0.012	0.581±0.091	**0.75**±0.001
	$\overline{\text{Recall}}$	**1.0**±0.0	**1.0**±0.0	0.894±0.025	0.917±0.025	0.584±0.129
	$\overline{\text{FP}}$	3.054±0.352	2.636±0.258	9.147±0.346	5.023±1.59	**1.5**±0.017
	$\overline{\text{Sparsity}}$	2.533±0.217	2.323±0.162	6.351±0.207	3.7±0.605	**1.559**±0.096
	$\overline{\text{NRMSE}}$	0.0027±0.0003	**0.0025**±0.0002	/	0.4311±0.00182	0.4312±0.0029
0.07	$\overline{\text{F2-score}}$	0.894±0.014	**0.895**±0.012	0.828±0.011	0.797±0.025	0.684±0.056
	$\overline{\text{Precision}}$	0.634±0.068	0.698±0.014	0.629±0.011	0.592±0.033	**0.754**±0.009
	$\overline{\text{Recall}}$	**0.985**±0.015	0.975±0.021	0.893±0.035	0.867±0.028	0.543±0.069
	$\overline{\text{FP}}$	2.759±0.573	2.15±0.139	9.253±0.642	4.82±0.383	**1.483**±0.11
	$\overline{\text{Sparsity}}$	2.232±0.295	1.957±0.13	6.39±0.396	3.566±0.112	**1.524**±0.123
	$\overline{\text{NRMSE}}$	0.015±0.003	**0.013**±0.001	/	0.433±0.002	0.43±0.004
0.1	$\overline{\text{F2-score}}$	0.89±0.022	**0.889**±0.023	0.825±0.007	0.784±0.019	0.668±0.064
	$\overline{\text{Precision}}$	0.646±0.049	0.719±0.03	0.606±0.009	0.604±0.023	**0.76**±0.013
	$\overline{\text{Recall}}$	**0.967**±0.026	0.961±0.017	0.91±0.021	0.832±0.016	0.526±0.038
	$\overline{\text{FP}}$	2.724±0.557	2.034±0.085	9.723±0.448	4.633±0.273	**1.467**±0.185
	$\overline{\text{Sparsity}}$	2.202±0.313	1.859±0.026	6.536±0.286	3.364±0.025	**1.491**±0.133
	$\overline{\text{NRMSE}}$	/	**0.227**±0.292	/	0.437±0.001	0.436±0.002

Table B.3: Lotka-Volterra Data-set: Comparison of the models over noise levels on interpretability metrics, F2-score, Precision, Recall, FP, Sparsity and prediction metric, NRMSE. The mean of metric X, denoted as $\overline{X}$, and standard deviation are computed over 50 runs. The winning method is shown in bold, and the second is underlined. The overflow values are replaced with "/".

SIR

Noise	Metric	SINDy	DaD_sub$^+$	kBest	RFE	SelectFI
0.001	$\overline{\text{F2-score}}$	0.948±0.007	**0.949**±0.007	0.58±0.003	0.536±0.011	0.57±0.014
	$\overline{\text{Precision}}$	0.843±0.015	**0.846±0.017**	0.42±0.011	0.391±0.061	0.649±0.005
	$\overline{\text{Recall}}$	1.0±0.0	**1.0±0.0**	0.665±0.003	0.585±0.04	0.41±0.017
	$\overline{\text{FP}}$	3.052±0.615	2.95±0.642	15.407±1.687	3.958±1.267	**0.171**±0.086
	$\overline{\text{Sparsity}}$	3.529±0.341	3.466±0.354	14.495±1.14	3.941±0.679	**1.736**±0.333
	$\overline{\text{NRMSE}}$	0.0013±0.0017	**0.0001**±0.0001	4.2456±0.2029	0.5743±0.0134	0.7902±0.0159
0.003	$\overline{\text{F2-score}}$	0.948±0.006	**0.958**±0.003	0.583±0.004	0.529±0.014	0.569±0.016
	$\overline{\text{Precision}}$	0.843±0.014	**0.868**±0.008	0.425±0.013	0.373±0.057	0.648±0.004
	$\overline{\text{Recall}}$	1.0±0.0	**1.0**±0.0	0.666±0.001	0.581±0.043	0.409±0.02
	$\overline{\text{FP}}$	3.05±0.566	1.995±0.352	14.058±1.322	4.247±1.213	**0.182**±0.087
	$\overline{\text{Sparsity}}$	3.532±0.309	2.955±0.244	13.666±0.918	4.024±0.645	**1.745**±0.348
	$\overline{\text{NRMSE}}$	0.004±0.0055	**0.0003**±0.0002	4.4057±0.0118	0.5699±0.0078	0.7911±0.0161
0.007	$\overline{\text{F2-score}}$	0.954±0.005	**0.97**±0.002	0.585±0.004	0.53±0.014	0.567±0.015
	$\overline{\text{Precision}}$	0.859±0.018	**0.897**±0.005	0.428±0.013	0.376±0.064	0.642±0.004
	$\overline{\text{Recall}}$	0.999±0.001	**1.0**±0.0	0.666±0.001	0.582±0.042	0.406±0.021
	$\overline{\text{FP}}$	2.658±0.577	1.145±0.124	13.151±1.188	4.138±1.539	**0.242**±0.103
	$\overline{\text{Sparsity}}$	3.401±0.321	2.487±0.085	13.196±0.778	3.975±0.846	**1.753**±0.34
	$\overline{\text{NRMSE}}$	0.127±0.178	**0.069**±0.097	5.321±0.142	0.571±0.01	0.788±0.006
0.01	$\overline{\text{F2-score}}$	0.957±0.007	**0.973**±0.003	0.587±0.003	0.522±0.012	0.567±0.015
	$\overline{\text{Precision}}$	0.865±0.017	**0.905**±0.008	0.434±0.01	0.37±0.059	0.637±0.005
	$\overline{\text{Recall}}$	1.0±0.0	**1.0**±0.0	0.665±0.003	0.563±0.049	0.413±0.018
	$\overline{\text{FP}}$	2.302±0.526	0.925±0.018	12.287±0.933	4.127±1.398	**0.293**±0.125
	$\overline{\text{Sparsity}}$	3.18±0.268	2.338±0.011	12.454±0.519	3.896±0.753	**1.79**±0.363
	$\overline{\text{NRMSE}}$	/	**0.003**±0.002	5.945±0.031	0.569±0.01	0.779±0.002

Table B.4: SIR Data-set: Comparison of the models over noise levels on interpretability metrics, F2-score, Precision, Recall, FP, Sparsity and prediction metric, NRMSE. The mean of metric X, denoted as $\overline{X}$, and standard deviation are computed over 50 runs. The winning method is shown in bold, and the second is underlined.

B.2.3 Effect of Data-set size on Interpretability

The data presented in Tables B.5 and B.6 provide a comparative analysis of the performance of various methods under different data-set sizes. The tables demonstrate that, compared to other methods, DaD_sub$^+$ consistently exhibits superior performance across all data-set sizes.

In fact, DaD_sub$^+$ consistently secures a position within the top two ranks in all scenarios. This consistent high-ranking performance of DaD_sub$^+$, irrespective of the data-set sizes, underscores its robustness and effectiveness.

Lotka-Volterra

Data-set Size	Metric	SINDy	DaD_sub^{+}	kBest	RFE	SelectFI
2	$\overline{\text{F2-score}}$	0.892±0.027	**0.894**±0.027	0.822±0.003	0.807±0.017	0.762±0.025
	$\overline{\text{Precision}}$	0.689±0.040	0.692±0.035	0.616±0.006	0.636±0.021	**0.758**±0.011
	$\overline{\text{Recall}}$	0.977±0.029	**0.979**±0.026	0.885±0.019	0.864±0.020	0.657±0.075
	$\overline{\text{FP}}$	2.321±0.235	2.222±0.193	9.142±0.479	3.932±0.507	**1.422**±0.101
	$\overline{\text{Sparsity}}$	2.185±0.351	2.094±0.272	6.272±0.248	3.194±0.303	**1.647**±0.027
	$\overline{\text{NRMSE}}$	0.011±0.018	**0.009**±0.278	/	0.43±0.004	0.429±0.004
5	$\overline{\text{F2-score}}$	0.892±0.026	**0.904**±0.019	0.826±0.006	0.801±0.024	0.668±0.008
	$\overline{\text{Precision}}$	0.659±0.057	0.701±0.034	0.621±0.017	0.626±0.057	**0.755**±0.005
	$\overline{\text{Recall}}$	**0.990**±0.013	0.985±0.019	0.894±0.012	0.858±0.063	0.509±0.007
	$\overline{\text{FP}}$	2.699±0.346	2.242±0.251	9.445±0.413	4.162±1.239	**1.470**±0.036
	$\overline{\text{Sparsity}}$	2.442±0.248	2.184±0.284	6.462±0.200	3.230±0.644	**1.482**±0.025
	$\overline{\text{NRMSE}}$	/	**0.007**±0.012	/	0.433±0.003	0.433±0.003
15	$\overline{\text{F2-score}}$	0.885±0.034	**0.909**±0.018	0.842±0.011	0.786±0.024	0.645±0.027
	$\overline{\text{Precision}}$	0.624±0.084	0.703±0.060	0.635±0.014	0.565±0.062	**0.747**±0.003
	$\overline{\text{Recall}}$	**0.997**±0.005	0.994±0.010	0.940±0.023	0.869±0.056	0.501±0.002
	$\overline{\text{FP}}$	3.079±0.523	2.301±0.466	10.102±0.509	5.060±1.336	**1.572**±0.068
	$\overline{\text{Sparsity}}$	2.639±0.151	2.223±0.298	6.913±0.357	3.567±0.572	**1.457**±0.059
	$\overline{\text{NRMSE}}$	/	**0.007**±0.045	/	0.433±0.002	0.433±0.002

Table B.5: Lotka-Volterra Data-set: Comparison of the models over data-set sizes on interpretability metrics, F2-score, Precision, Recall, FP, Sparsity and prediction metric, NRMSE. The mean of metric X, denoted as $\overline{X}$, and standard deviation are computed over 50 runs. The winning method is shown in bold, and the second is underlined. The overflow values are replaced with "/".

SIR

Data-set Size	Metric	SINDy	DaD_sub^{+}	kBest	RFE	SelectFI
5	$\overline{\text{F2-score}}$	0.959±0.005	**0.965**±0.007	0.587±0.004	0.543±0.006	0.585±0.001
	$\overline{\text{Precision}}$	0.871±0.013	**0.884**±0.017	0.437±0.007	0.446±0.01	0.639±0.005
	$\overline{\text{Recall}}$	0.999±0.001	**1.0**±0.0	0.663±0.002	0.529±0.016	0.429±0.003
	$\overline{\text{FP}}$	2.116±0.328	1.507±0.562	12.567±1.185	2.58±0.207	**0.335**±0.077
	$\overline{\text{Sparsity}}$	3.059±0.134	2.671±0.315	12.71±0.717	3.121±0.132	**2.135**±0.038
	$\overline{\text{NRMSE}}$	/	**0.036**±0.068	5.048±0.779	0.578±0.004	0.794±0.01
10	$\overline{\text{F2-score}}$	0.949±0.004	**0.962**±0.012	0.584±0.002	0.527±0.006	0.565±0.002
	$\overline{\text{Precision}}$	0.949±0.004	**0.962**±0.012	0.584±0.002	0.527±0.006	0.565±0.002
	$\overline{\text{Recall}}$	1.0±0.0	**1.0**±0.0	0.667±0.0	0.596±0.008	0.408±0.006
	$\overline{\text{FP}}$	2.995±0.377	1.83±1.02	13.548±1.063	4.697±0.324	**0.188**±0.049
	$\overline{\text{Sparsity}}$	3.533±0.192	2.86±0.582	13.322±0.638	4.339±0.122	**1.676**±0.021
	$\overline{\text{NRMSE}}$	/	0.937±0.107	5.067±0.82	0.565±0.001	**0.785**±0.001
15	$\overline{\text{F2-score}}$	0.948±0.005	**0.96**±0.014	0.58±0.003	0.518±0.007	0.556±0.001
	$\overline{\text{Precision}}$	0.842±0.011	**0.873**±0.033	0.415±0.009	0.332±0.004	0.648±0.006
	$\overline{\text{Recall}}$	1.0±0.0	**1.0**±0.0	0.667±0.0	0.608±0.014	0.391±0.003
	$\overline{\text{FP}}$	3.185±0.38	2.027±1.104	15.062±1.781	5.075±0.078	**0.143**±0.044
	$\overline{\text{Sparsity}}$	3.639±0.176	2.973±0.592	14.326±1.231	4.417±0.073	**1.457**±0.014
	$\overline{\text{NRMSE}}$	0.001±0.002	**0.001**±0.001	4.911±0.824	0.564±0.001	0.78±0.003

Table B.6: SIR Data-set: Comparison of the models over data-set sizes on interpretability metrics, F2-score, Precision, Recall, FP, Sparsity and prediction metric, NRMSE. The mean of metric X, denoted as $\overline{X}$, and standard deviation are computed over 50 runs. The winning method is shown in bold, and the second is underlined.

APPENDIX C

Introduction en Francais

C.1 Contexte

Les années 1960 ont vu l'émergence de l'intelligence artificielle (IA) en tant que domaine distinct. Les chercheurs pionniers Allen Newell et Herbert Simon visaient à reproduire les capacités de résolution de problèmes humains à travers le développement du programme "Logic Theorist" basé sur la logique (Russell & Norvig, 2010).

Les premiers modèles, fondés sur des opérateurs logiques, le raisonnement symbolique, des structures arborescentes et des systèmes basés sur des règles, ont conduit au développement de programmes informatiques tels que les règles logiques, les systèmes experts et les arbres de décision. Ces modèles étaient relativement simples, ce qui facilitait la compréhension des processus de prise de décision.

Bien que ces modèles se soient révélés prometteurs dans des domaines spécifiques, leurs limites sont devenues évidentes face à des défis plus complexes. Parallèlement, des efforts ont été faits pour développer des approches plus sophistiquées, y compris les premiers réseaux neuronaux comme le perceptron. Cependant, ces tentatives ont été limitées par diverses contraintes techniques, entravant finalement leur succès.

Depuis la fin du 20e siècle, des avancées significatives en matière de matériel informatique ont considérablement augmenté la puissance de calcul. Cela, associé au développement simultané de la capacité de stockage des données, a permis la collecte et le stockage de grandes quantités de données. Ces progrès techniques, combinés aux avancées rapides des algorithmes et de la recherche mathématique, ont posé les bases de la popularisation de l'intelligence artificielle.

Le tournant de l'IA a eu lieu lors de la compétition ImageNet en 2012, avec le succès de l'apprentissage profond dans les tâches de vision par ordinateur (Krizhevsky et al., 2012). Depuis, l'apprentissage automatique a fait des progrès significatifs, produisant des modèles de plus en plus efficaces dans diverses tâches, allant de la prise de décision et de la prédiction à la prévision. Son influence s'étend au-delà des IA conversationnelles via le traitement du langage naturel, de la vision par ordinateur pour la reconnaissance et la génération d'images et de vidéos, de l'analyse vocale et des systèmes de recommandation. En conséquence, les applications de l'apprentissage automatique ont pénétré tous les secteurs, y compris la santé, la finance, l'industrie, l'automobile et le marketing. En effet, l'adoption de l'IA est devenue stratégiquement impérative pour les entreprises afin de rester compétitives et efficaces.

Bien que les avancées aient permis aux modèles d'apprentissage automatique d'atteindre une

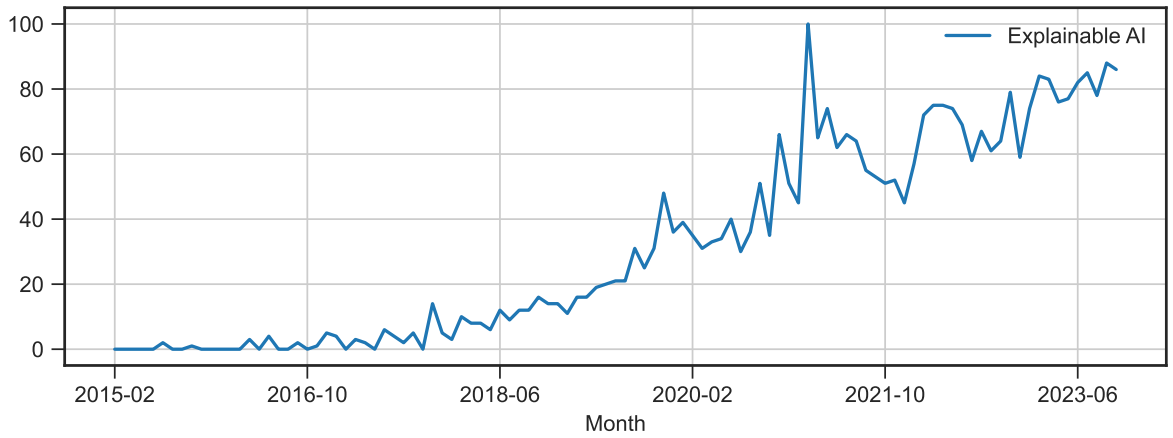


Figure S1: Indice Google Trends (valeur maximale de 100) du terme "Explainable AI" au cours des dernières années (2015–2023).

précision et une efficacité impressionnantes, elles ont également produit des systèmes de plus en plus complexes. Souvent qualifiés de "boîtes noires" ou de modèles "opaques" - des modèles aux fonctionnements internes complexes ou inconnus où seuls les entrées et sorties sont observées - par opposition aux modèles transparents, leurs conceptions entravent la compréhension et la justification de leurs décisions lorsqu'ils sont appliqués dans un environnement réel.

L'opacité des modèles d'IA amplifie les considérations éthiques concernant les biais dans les algorithmes pendant l'apprentissage et la responsabilité des décisions, en particulier dans des domaines critiques comme la santé (Morley et al., 2020), les véhicules autonomes (Martinho et al., 2021), le recrutement (Hofeditz et al., 2022) et les chatbots (Wiltz, 2017). Ce manque de transparence entrave notre compréhension de la manière dont le modèle arrive à ses décisions, rendant difficile l'identification et la correction des biais ou des faiblesses qui pourraient conduire à des erreurs et des résultats discriminatoires. Par conséquent, les modèles opaques soulèvent des préoccupations éthiques et juridiques pour les entreprises et la société.

L'adoption généralisée de l'IA, en particulier dans les domaines sensibles impactant les humains, la société et les finances, dépend fortement de la confiance et de l'acceptation. Cela se traduit par un besoin clair de la part des différents acteurs (Preece et al., 2018), motivés par trois exigences principales : la transparence, l'interaction humaine et des modèles dignes de confiance (Vilone & Longo, 2021).

La nécessité de comprendre les systèmes d'IA a conduit à l'émergence de l'Intelligence Artificielle Explicable (XAI). Ce domaine se concentre sur le développement et l'analyse de modèles d'apprentissage automatique, nouveaux et existants, avec les objectifs suivants (Barredo Arrieta et al., 2020) :

- ❖ Identifier et corriger les biais
- ❖ Améliorer la robustesse des modèles
- ❖ Fournir une explication causale des décisions prises par les modèles

C.2 Pourquoi avons-nous besoin d'explications

L'apprentissage automatique offre un moyen puissant d'améliorer les capacités prédictives des systèmes, leur permettant de prendre des décisions éclairées basées sur des informations issues des données. Cela est réalisé en utilisant divers algorithmes, chacun fournissant un certain degré d'explication, tel que l'incertitude, qui peut servir d'indicateur de la confiance algorithmique. Cependant, dans des contextes où les conséquences des erreurs peuvent être graves, comme dans des domaines critiques tels que la santé ou les véhicules autonomes, le manque inhérent de transparence et la difficulté à comprendre le processus de prise de décision deviennent une préoccupation majeure.

La fiabilité d'un modèle d'apprentissage automatique est intimement liée à sa capacité à expliquer ses décisions. Dans ces scénarios à enjeux élevés, il est impératif de disposer de modèles qui soient non seulement prédictifs et précis, mais aussi interprétables. Cela permettra aux parties prenantes de comprendre comment et pourquoi une décision particulière a été prise, offrant ainsi transparence et responsabilité. Dans ces contextes, l'interprétabilité devient une nécessité éthique et pratique.

Dans de telles applications, il est nécessaire d'avoir des garanties non seulement sur les performances du modèle, mais aussi sur sa fiabilité. Il est essentiel de développer des méthodes pour évaluer rigoureusement les performances et la robustesse sous diverses conditions de données et scénarios, ainsi que les mécanismes internes et les processus de prise de décision du modèle, afin de construire cette confiance dans les modèles d'apprentissage automatique. Des métriques quantitatives sont généralement utilisées pour l'évaluation, mais des évaluations qualitatives peuvent également être incluses en fonction du problème et des objectifs.

C.3 Définitions utilisée dans cette thèse

L'efficacité des méthodes de XAI dépend fortement du contexte d'application. Le domaine, les objectifs de l'utilisateur, ainsi que les tâches spécifiques influencent de manière déterminante le choix de l'approche la plus appropriée.

L'intelligence artificielle explicable est un concept qui regroupe l'ensemble des approches visant à rendre les modèles compréhensibles, de l'analyse des relations entre les données à la structure fondamentale et au processus de prise de décision d'un modèle d'apprentissage automatique. L'objectif est de permettre une compréhension des mécanismes internes, de la théorie sous-jacente et du processus de prise de décision qui conduisent à un résultat, tout en communiquant ces informations de manière claire et justifiée. Ce domaine a émergé parallèlement à l'augmentation du nombre de modèles dits "boîtes noires" et vise à répondre à leur complexité croissante. Le développement de la XAI est devenu indispensable en raison des nouvelles régulations et de la nécessité de renforcer la confiance des utilisateurs dans les systèmes d'intelligence artificielle.

Malgré l'importance croissante de l'intelligence artificielle explicable (XAI), il n'existe pas encore de consensus sur les définitions et les objectifs précis de ce domaine. Les chercheurs de diverses disciplines ont proposé des définitions et des taxonomies variées, ce qui a conduit à une multiplicité d'approches et de méthodologies, chacune étant influencée par des perspectives et des finalités spécifiques. En l'absence d'un cadre théorique unifié, évaluer et comparer les modèles devient une tâche complexe, risquant ainsi de freiner les progrès dans ce domaine (Nguyen & Martínez, 2020).

Face à cette diversité de perspectives, il est essentiel de clarifier les défis, l'état actuel du domaine, ainsi que les définitions retenues dans ce travail pour en délimiter le cadre. Dans cette thèse, afin de garantir la clarté et la cohérence, les termes "interprétabilité" et "explicabilité" seront compris selon les définitions suivantes :

Definition C.3.1 (Interprétabilité)**.** *L'**interprétabilité** est définie comme la capacité d'un modèle à décrire son processus de prise de décision de manière compréhensible pour un utilisateur. Les modèles possédant cette caractéristique sont qualifiés d'interprétables.*

Definition C.3.2 (Explicabilité)**.** *L'**explicabilité** est définie comme la capacité à fournir une explication des raisons sous-jacentes à une décision d'un modèle, de manière intelligible pour un utilisateur. Les modèles répondant à ce critère sont qualifiés d'explicables (Barredo Arrieta et al., 2020).*

C.4 Objectif de mes recherches

L'un des aspects clés de cette recherche consiste à comprendre les variations temporelles et à identifier les causes des déviations par rapport aux conditions normales de fonctionnement ou à mettre en évidence les dynamiques cachées dans les données de séries temporelles. En effet, les données de séries temporelles suivent essentiellement l'évolution de variables au cours du temps, comme les prix des actions, les données météorologiques, la surveillance cérébrale ou le suivi du rythme cardiaque. En identifiant les motifs dans ces données, nous pouvons mieux comprendre l'interaction des variables, améliorer les prévisions, approfondir notre compréhension de la stabilité des systèmes et développer des stratégies d'intervention efficaces.

Objectif 1 : Développer un algorithme permettant d'identifier les causes des comportements normaux et anormaux dans une série temporelle.

- ❖ Cet objectif vise à développer un algorithme capable de fournir des informations et des explications sur les déviations par rapport au comportement normal.
- ❖ L'algorithme doit être en mesure de traiter divers types de séries temporelles, y compris les données numériques, catégorielles et multivariées.
- ❖ Les explications fournies par l'algorithme doivent être compréhensibles par les humains et révéler les causes sous-jacentes.

Un autre aspect clé de cette recherche réside dans la résolution des lacunes importantes dans l'analyse des données de séries temporelles. Premièrement, nous manquons de compréhension approfondie des causes et du moment des comportements anormaux, ce qui limite notre capacité à anticiper et à atténuer les risques potentiels. Deuxièmement, nous avons des difficultés à expliquer le comportement dynamique des systèmes et à identifier les variables qui influencent leur évolution. Ce manque de pouvoir explicatif restreint notre capacité à comprendre les systèmes et à optimiser leurs performances.

Objectif 2 : Développer un modèle de prévision qui révèle la dynamique sous-jacente d'une série temporelle et prédit son évolution future.

- ❖ Cet objectif vise à développer un modèle de prévision capable non seulement de prédire les valeurs futures d'une série temporelle, mais aussi d'expliquer ses prédictions.
- ❖ Le modèle doit être transparent, permettant aux utilisateurs de comprendre les facteurs influençant la prévision et de tirer des enseignements sur les tendances et motifs sous-jacents.
- ❖ Le modèle doit pouvoir gérer des séries temporelles complexes comportant des non-linéarités.

C.5 Plan de la Thèse

Cette section présente la structure de la thèse, détaillant chaque chapitre et ses contributions spécifiques.

Chapitre 2 : Comprendre les Modèles de Machine Learning Ce chapitre établit les bases de la recherche en définissant et contextualisant l'Intelligence Artificielle Explicable (XAI). Nous explorons les défis, les définitions, les méthodes et les évaluations de ce domaine. Une revue de la littérature détaillée couvre divers modèles, incluant les modèles de régression, les modèles basés sur des règles, les réseaux bayésiens, la régression symbolique et les méthodes interprétables locales. Les sections 2.1 à 2.2 présentent de manière générale le domaine de l'IA explicable, en incluant un historique détaillé de la thématique, les impacts récents dans la législation (RGPD et AI Act) et les principales questions adressées dans ce domaine : compréhension du modèle et de ses limites, et ce qu'il peut nous apprendre en termes de modélisation. Une taxonomie des approches XAI est présentée en section 2.3 pour structurer les différentes méthodes selon des critères tels que leur portée globale ou locale, leur caractère post hoc ou intégré, ainsi que leur indépendance par rapport aux modèles. Cette classification permet de mieux situer chaque approche dans son contexte d'application spécifique. La section 2.4 se concentre sur les différentes méthodes développées autour de trois questions centrales: explicabilité, interprétabilité et causalité, avec une présentation pédagogique de chaque approche. Enfin, la section 2.5 aborde la question cruciale de l'évaluation de l'explicabilité, en considérant des dimensions telles que la clarté, la simplicité et le caractère général.

Chapitre 3 : Analyse des Séries Temporelles : Modélisation et Identification des Facteurs d'Influence Ce chapitre se concentre sur l'analyse des séries temporelles, en particulier à l'analyse des causes premières (Root Cause Analysis) et aux limites des méthodes actuelles.

La section 3.1 presente une introduction détaillée à l'analyse des causes premières, en précisant la terminologie, les familles d'approches, ainsi que les types d'anomalies identifiés dans ce contexte. La section 3.2 expose les défis spécifiques à ce domaine, en formulant les questions clés auxquelles les chercheurs doivent répondre, tout en identifiant les principales limitations des méthodes existantes. Cette section propose ensuite une définition du problème étudié.

La section 3.3 présente un état de l'art des approches disponibles dans ce domaine, couvrant les méthodes statistiques, l'extraction de règles, le machine learning, ainsi que les méthodes causales. Cette revue est accompagnée d'exemples et de la presentations de certaines limitations de ces approaches ces approches. Enfin, la section 3.4 situe notre travail dans le contexte de ces recherches, en détaillant nos contributions spécifiques et en justifiant la nécessité de développer

des modèles d'analyse causale des règles d'association plus fiables et transparents pour les séries temporelles.

Chapitre 4 : Règles Causales et Interprétables pour l'Analyse des Séries Temporelles
Ce chapitre présente notre première contribution à l'analyse des causes premières. De nombreux modèles complexes permettent d'apprendre à partir de données labelisées et de prédire la survenue d'événements, tels que des défaillances, à partir d'un ensemble de données de séries temporelles contenant un grand nombre de variables. Bien qu'une grande précision puisse être atteinte, certaines erreurs et événements manqués peuvent potentiellement entraîner des pertes importantes en raison du manque de prise de décision et de compréhension du modèle. Nos premiers travaux ont relevé ce défi en proposant une approche interprétable pour découvrir la cause première des défaillances.

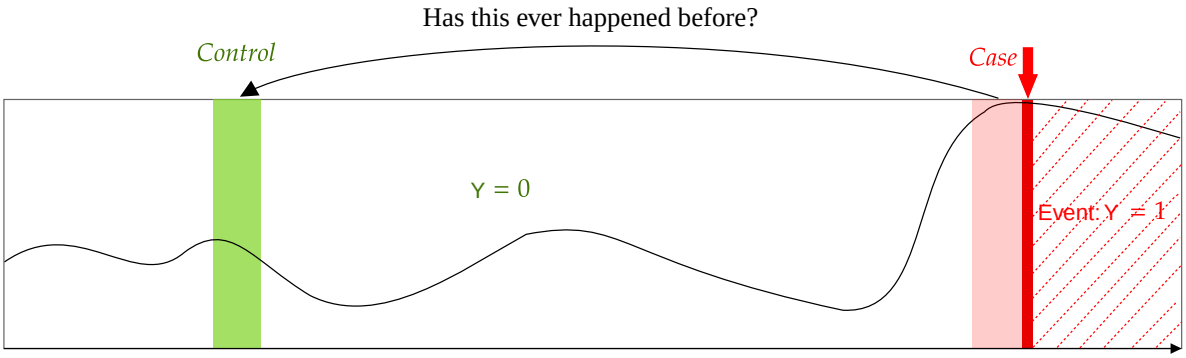


Figure S2: Schéma de base du design de crossover de cas avec une seule période de contrôle et une seule période de cas.

Notre approche vise à construire des règles simples en utilisant un cadre de fouille de règles d'association. La première étape consiste à transformer et discrétiser l'ensemble de données pour créer une nouvelle base de données destinée à l'inférence. La deuxième étape implique l'incorporation de la causalité, car la fouille de règles d'association ne traite que des associations. Pour cela, nous utilisons une approche épidémiologique appelée le case-crossover design. Cette méthode, largement utilisée en épidémiologie pour comprendre les origines de phénomènes apparaissant soudainement (comme les crises cardiaques, les accidents ou les blessures (Estberg et al., 1998; Maclure & Mittleman, 1997; Mittleman et al., 1993, 1995)), permet d'établir une relation causale entre une exposition et un événement en démontrant que la survenue de l'exposition provoque effectivement l'événement. La combinaison de ces deux approches fournit ainsi des règles causales et interprétables pour comprendre les causes des défaillances.

Ensuite, deux algorithmes prédictifs ont été construits sur la base des règles causales, permettant de prévoir la survenue des défaillances. Le premier algorithme sélectionne des règles en fonction de plusieurs critères et agrège les décisions des règles pour faire une prédiction globale. Le second algorithme améliore le premier en ajoutant des anti-règles, qui sont des règles prédisant une situation normale. L'agrégation des deux types de règles permet d'obtenir un algorithme prédictif puissant.

Cette approche a été testée sur un ensemble de données réel où un phénomène appelé engorgement se produit brièvement dans le temps et induit une défaillance. L'objectif était

d'identifier les causes du problème et de fournir aux opérateurs des informations simples et interprétables. L'approche et les algorithmes prédictifs ont été appliqués à cet ensemble de données et ont montré de bonnes performances, validées par des experts du domaine.

Chapitre 5 : Prévision Multi-horizon et Découverte de Systèmes Dynamiques Ce chapitre explore le domaine de la prévision multi-horizon et de la découverte de systèmes dynamiques.

La section 5.1 introduit le domaine de la modélisation dynamique, en définissant les concepts clés et en présentant les différents types de modèles dynamiques. La section 5.2 offre une vue d'ensemble des méthodologies de prévision des séries temporelles, en se concentrant particulièrement sur les modèles paramétriques appliqués aux séries temporelles multivariées et leur utilisation pour la prévision.

La section 5.3 développe l'idée d'utiliser la prévision des séries temporelles comme un outil pour la découverte dynamique, en tenant compte des contraintes d'interprétabilité et de la gestion d'un grand nombre de composantes. Nous y abordons les défis associés à la modélisation des processus dynamiques sous-jacents aux séries temporelles complexes, et mettons en lumière l'efficacité des approches de régression pénalisée pour surmonter ces difficultés.

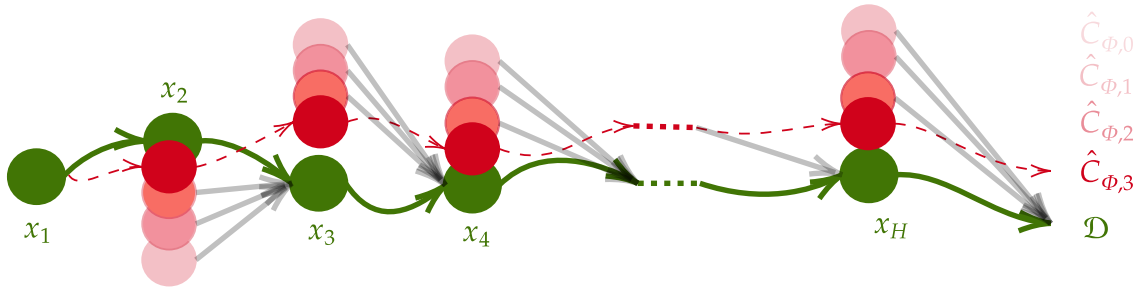
La section 5.4 présente un état de l'art des méthodes utilisées en modélisation dynamique, en détaillant les approches existantes et leurs limitations. Enfin, la section 5.5 discute de l'intégration de notre travail dans le cadre des recherches actuelles, en soulignant nos contributions spécifiques et la manière dont elles enrichissent le domaine de la modélisation dynamique et de la prévision multi-horizon.

Chapitre 6 : Modèle de Prévision Cohérent et Interprétable Après avoir fourni les outils pour analyser les causes de phénomènes spécifiques à partir de données de séries temporelles multivariées potentiellement de haute dimension, nous avons abordé le problème sous une perspective globale, en cherchant à comprendre les dynamiques sous-jacentes qui génèrent ces séries temporelles. L'objectif principal était d'extraire une équation à partir des observations, permettant de déterminer les relations entre les variables explicatives et une variable de sortie. Cette équation generée à partir d'un modèle interprétable, offre une meilleure compréhension des phénomènes en jeu et des systèmes sous-jacents.

Dans ce cadre, nous avons développé une approche qui tire parti des méthodes interprétables, telles que les modèles vectoriels autorégressifs, pour apprendre les dynamiques sous-jacentes. Pour capturer la nature non linéaire des données réelles, les dimensions d'entrée ont été augmentées en introduisant des non-linéarités dans les variables d'entrée. Deux principaux défis ont été identifiés : la haute dimensionnalité des données et la propagation d'erreurs dans un cadre de prévision multi-horizon.

Pour traiter le problème de la dimensionnalité élevée, nous avons appliqué des méthodes de régression pénalisée favorisant la parcimonie, ce qui permet de sélectionner un ensemble restreint de variables significatives, réduisant ainsi la complexité du modèle. Pour surmonter le problème de la propagation des erreurs, nous avons intégré une approche inspirée de l'algorithme DAgger (Dataset Aggregation), qui améliore l'efficacité du modèle en utilisant de manière optimale les données d'entraînement. L'algorithme itératif proposé apprend un ensemble de modèles candidats, corrigeant successivement les trajectoires pour mieux capturer les dynamiques sous-jacentes.

Figure S3: La figure illustre la capacité de l'algorithme à apprendre et à corriger itérativement les trajectoires. L'objectif est de découvrir la dynamique sous-jacente réelle représentée par la distribution inconnue $\mathcal{D}$ à partir des séries temporelles observées en vert. Initialement, l'algorithme apprend un modèle récursif, $\mathcal{C}\Phi, 0$, qui propage les erreurs. Lors des itérations suivantes, l'algorithme utilise les trajectoires précédentes pour augmenter les données d'entraînement afin d'améliorer progressivement l'apprentissage de la dynamique réelle par le modèle. Ici, la figure représente le processus d'apprentissage jusqu'à la troisième itération, où le modèle final est désigné par $\mathcal{C}\Phi, 3$.



Cette méthodologie présente à la fois des contributions méthodologiques, théoriques et expérimentales. Elle a été testée sur des séries temporelles synthétiques dérivées d'équations différentielles ordinaires (ODE) pour retrouver les équations de l'ODE discrétisée. Après validation à l'aide de métriques de performance de prévision et d'analyses d'interprétabilité, nous avons appliqué notre modèle à un ensemble de données éoliennes.

C.6 Perspectives

Cette thèse pose les bases pour des recherches futures. Nous discutons des défis et des opportunités pour progresser davantage grâce aux deux approches développées.

Algorithme de règles causales

- **Amélioration de la discrétisation et de la labellisation des séries temporelles** L'amélioration de la discrétisation des séries temporelles et de la labellisation des données est cruciale. L'utilisation d'algorithmes comme SAX peut réduire la perte d'information. De plus, des évaluations supplémentaires sur des ensembles de données synthétiques sont nécessaires pour déterminer l'efficacité de l'algorithme.
- **Causalité de Granger et Interventions** L'analyse causale de Granger repose sur des hypothèses spécifiques qui peuvent introduire des corrélations fallacieuses. Une évaluation critique des hypothèses de notre algorithme est nécessaire pour garantir l'identification précise des effets causaux. De plus, identifier les variables manipulables dans le graphe causal permettrait de développer des stratégies d'intervention pour prévenir les défaillances.

- **Prise en compte des variables non observées** Ajouter des variables non observées à l'analyse pourrait fournir des informations supplémentaires pour une compréhension globale du système.

Découverte de systèmes dynamiques

- **Limitation théorique** Il est nécessaire de déterminer si le système d'équations est identifiable et de considérer l'influence potentielle des variables latentes.
- **Étude approfondie du problème de multicolinéarité** La présence de multicolinéarité dans les ensembles de données peut empêcher une estimation précise des paramètres. Les techniques de régularisation comme Lasso doivent être utilisées avec précaution pour garantir l'identification du vrai système.
- **Alternative au Lasso** Explorer des alternatives à l'algorithme Lasso, telles que la régression régulière détendue et éparse (SR3), pourrait améliorer l'efficacité des calculs dans notre algorithme de découverte dynamique.

Conclusion et Discussion

Cette thèse fourni des outils pour analyser les causes de phénomènes spécifiques à partir de données de séries temporelles multivariées et pour apprendre les dynamiques sous-jacentes générant les séries temporelles. Les perspectives futures incluent l'amélioration de la discrétisation et de la labellisation des séries temporelles, la prise en compte des variables non observées, la résolution du problème de multicolinéarité et l'exploration de méthodes de régression pénalisée alternatives. Développer des méthodes de détection des anomalies en temps réel et renforcer l'interaction entre les modèles explicables et les utilisateurs pour améliorer la prise de décision sont également des axes de recherche importants.

C.7 Contributions

Cette thèse apporte des contributions dans le domaine de l'intelligence artificielle explicable (XAI) appliquée aux séries temporelles multivariées. Les principales contributions sont les suivantes :

Article KDD : Règles causales et interprétables pour l'analyse des séries temporelles

L'une des contributions majeures de cette thèse est le développement d'un algorithme pour l'analyse des causes premières dans les séries temporelles, combinant le design de crossover de cas avec l'algorithme Apriori de fouille de règles d'association. Cette méthode innovante permet d'identifier les causes des anomalies en temps réel et de fournir des explications interprétables. Les résultats de cette recherche ont été publiés dans l'article de la conférence KDD intitulé "Causal and interpretable rules for time series analysis".

Brevet : procédé de contrôle d'un système et produit programme d'ordinateur associé

La thèse a conduit au dépôt d'un brevet intitulé "Procédé de contrôle d'un système et produit programme d'ordinateur associé", en collaboration avec TotalEnergies OneTech, le Centre

National de la Recherche Scientifique et l'École Polytechnique. Ce brevet, numéro FR3124868B1, décrit une méthode innovante pour le contrôle de systèmes basés sur l'analyse des séries temporelles, démontrant l'application pratique et industrielle des travaux de recherche. Les détails du brevet sont disponibles sous le numéro d'enregistrement national 21 07171, 2023 ⟨hal-04455926⟩.

Code : Algorithmes CAPP L'étude causale a mené à l'élaboration de deux algorithmes CAPP (Case-crossover APriori) 1 et 2. Le code source de ces algorithmes ont été rendu public pour permettre la reproduction des résultats et encourager d'autres chercheurs à utiliser et améliorer cet outil. Le code est accessible à l'adresse suivante : https://github.com/amindh/CAPP.

Article : Framework de prévision multi-horizon interprétable et cohérent Une autre contribution majeure réside dans le développement d'un algorithme de prévision multi-horizon qui allie précision et interprétabilité. Inspiré par l'algorithme DAgger, cet algorithme propose des améliorations pour la prédiction à long terme et la découverte des dynamiques sous-jacentes des systèmes. Cette contribution a donné lieu à un article intitulé "Learning from mistakes: an Interpretable and Coherent Multi-step Ahead Time Series Forecasting Framework".

Code : Framework de prévision multi-horizon L'étude de prévision multi-horizon menée dans cette thèse a également abouti au développement d'un algorithme, dont le code source est accessible à l'adresse suivante : https://github.com/amindh/multi_step.

Ces contributions montrent l'impact des travaux de cette thèse sur la recherche académique et les applications industrielles, en proposant des solutions innovantes et pratiques pour l'analyse des séries temporelles et la prévision.

Bibliography

Acharki, N., Lugo, R., Bertoncello, A., and Garnier, J. Comparison of meta-learners for estimating multi-valued treatment heterogeneous effects. In *International Conference on Machine Learning*, pp. 91–132. PMLR, 2023. URL https://proceedings.mlr.press/v202/acharki23a.

Agarwal, A., Negahban, S., and Wainwright, M. J. Fast global convergence rates of gradient methods for high-dimensional statistical recovery. *Advances in Neural Information Processing Systems*, 23, 2010. URL https://proceedings.neurips.cc/paper/2010/hash/7cce53cf90577442771720a370c3c723-Abstract.html.

Agrawal, R. and Srikant, R. Fast algorithms for mining association rules. In *Proc. 20th int. conf. very large data bases, VLDB*, volume 1215, pp. 487–499. Santiago, 1994. URL https://www.it.uu.se/edu/course/homepage/infoutv/ht08/vldb94_rj.pdf.

Agrawal, R., Imieliński, T., and Swami, A. Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD international conference on Management of data*, pp. 207–216, Washington D.C. USA, June 1993. ACM. ISBN 978-0-89791-592-2. doi: 10.1145/170035.170072. URL https://dl.acm.org/doi/10.1145/170035.170072.

Akaike, H. A new look at the statistical model identification. *IEEE transactions on automatic control*, 19(6):716–723, 1974. URL https://ieeexplore.ieee.org/abstract/document/1100705/. Publisher: Ieee.

Andrews, R., Diederich, J., and Tickle, A. B. Survey and critique of techniques for extracting rules from trained artificial neural networks. *Knowledge-based systems*, 8(6):373–389, 1995. URL https://www.sciencedirect.com/science/article/pii/0950705196819204. Publisher: Elsevier.

Angelopoulou, A., Kapetanios, E., Smith, D. H., Steuber, V., Woll, B., and Zeller, F. Explanation in human-AI systems. *Frontiers in Artificial Intelligence*, 5:1048568, October 2022. ISSN 2624-8212. doi: 10.3389/frai.2022.1048568. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9644432/.

Assaad, C. K., Devijver, E., and Gaussier, E. Survey and Evaluation of Causal Discovery Methods for Time Series. *Journal of Artificial Intelligence Research*, 73:767–819, February 2022. ISSN 1076-9757. doi: 10.1613/jair.1.13428. URL https://www.jair.org/index.php/jair/article/view/13428.

Assaad, C. K., Ez-zejjari, I., and Zan, L. Root Cause Identification for Collective Anomalies in Time Series given an Acyclic Summary Causal Graph with Loops, October 2023. URL http://arxiv.org/abs/2303.04038. arXiv:2303.04038 [cs].

Ayodeji, A., Amidu, M. A., Olatubosun, S. A., Addad, Y., and Ahmed, H. Deep learning for safety assessment of nuclear power reactors: Reliability, explainability, and research opportunities. *Progress in Nuclear Energy*, 151:104339, September 2022. ISSN 0149-1970. doi: 10.1016/j.pnucene.2022.104339. URL https://www.sciencedirect.com/science/article/pii/S0149197022002141.

Azevedo, P. J. and Jorge, A. M. Comparing Rule Measures for Predictive Association Rules. In Kok, J. N., Koronacki, J., Mantaras, R. L. d., Matwin, S., Mladenič, D., and Skowron, A. (eds.), *Machine Learning: ECML 2007*, Lecture Notes in Computer Science, pp. 510–517, Berlin, Heidelberg, 2007. Springer. ISBN 978-3-540-74958-5. doi: 10.1007/978-3-540-74958-5_47.

Bach, F. R. Bolasso: model consistent Lasso estimation through the bootstrap. In *Proceedings of the 25th international conference on Machine learning - ICML '08*, pp. 33–40, Helsinki, Finland, 2008. ACM Press. ISBN 978-1-60558-205-4. doi: 10.1145/1390156.1390161. URL http://portal.acm.org/citation.cfm?doid=1390156.1390161.

Barker, J. Machine learning in M4: What makes a good unstructured model? *International Journal of Forecasting*, 36(1):150–155, January 2020. ISSN 01692070. doi: 10.1016/j.ijforecast.2019.06.001. URL https://linkinghub.elsevier.com/retrieve/pii/S0169207019301463.

Barredo Arrieta, A., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., Garcia, S., Gil-Lopez, S., Molina, D., Benjamins, R., Chatila, R., and Herrera, F. Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion*, 58:82–115, June 2020. ISSN 1566-2535. doi: 10.1016/j.inffus.2019.12.012. URL https://www.sciencedirect.com/science/article/pii/S1566253519308103.

Ben Taieb, S. *Machine learning strategies for multi-step-ahead time series forecasting*. PhD thesis, October 2014. URL http://hdl.handle.net/2013/. Publisher: Université libre de Bruxelles.

Ben Taieb, S., Bontempi, G., Atiya, A. F., and Sorjamaa, A. A review and comparison of strategies for multi-step ahead time series forecasting based on the NN5 forecasting competition. *Expert systems with applications*, 39(8):7067–7083, 2012. Publisher: Elsevier.

Bhatt, U., Xiang, A., Sharma, S., Weller, A., Taly, A., Jia, Y., Ghosh, J., Puri, R., Moura, J. M. F., and Eckersley, P. Explainable machine learning in deployment. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pp. 648–657, Barcelona Spain, January 2020. ACM. ISBN 978-1-4503-6936-7. doi: 10.1145/3351095.3375624. URL https://dl.acm.org/doi/10.1145/3351095.3375624.

Biran, O. and Cotton, C. Explanation and justification in machine learning: A survey. In *IJCAI-17 workshop on explainable AI (XAI)*, volume 8, pp. 8–13, 2017. URL http://www.cs.columbia.edu/~orb/papers/xai_survey_paper_2017.pdf. Issue: 1.

Bishop, C. M. *Pattern recognition and machine learning*. Information science and statistics. Springer, New York, 2006. ISBN 978-0-387-31073-2.

Blumensath, T. and Davies, M. E. Iterative hard thresholding for compressed sensing. *Applied and computational harmonic analysis*, 27(3):265–274, 2009. URL https://www.sciencedirect.com/science/article/pii/S1063520309000384. Publisher: Elsevier.

Bontempi, G. Long term time series prediction with multi-input multi-output local learning. *Proc. 2nd ESTSP*, pp. 145–154, 2008. URL https://www.academia.edu/download/51605610/Reliability_of_ARMA_and_GARCH_models_of_20170202-14409-1hvugmc.pdf#page=145.

Bontempi, G. Handbook statistical foundations of machine learning. *Self-published, Brüssel*, 2021.

Bontempi, G., Ben Taieb, S., and Le Borgne, Y.-A. Machine Learning Strategies for Time Series Forecasting. In Aufaure, M.-A. and Zimányi, E. (eds.), *Business Intelligence*, volume 138, pp. 62–77. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013. ISBN 978-3-642-36317-7 978-3-642-36318-4. doi: 10.1007/978-3-642-36318-4_3. URL https://link.springer.com/10.1007/978-3-642-36318-4_3. Series Title: Lecture Notes in Business Information Processing.

Both, G.-J. *Model Discovery of Partial Differential Equations*. PhD Thesis, Université Paris sciences et lettres, 2021. URL https://www.theses.fr/2021UPSLS088.

Bouche, D., Flamary, R., d'Alché Buc, F., Plougonven, R., Clausel, M., Badosa, J., and Drobinski, P. Wind power predictions from nowcasts to 4-hour forecasts: a learning approach with variable selection, December 2022. URL http://arxiv.org/abs/2204.09362. Issue: arXiv:2204.09362 arXiv:2204.09362 [cs, stat].

Breiman, L. Random Forests. *Machine Learning*, 45:5–32, 2001.

Breiman, L., Gordon, A. D., Friedman, J. H., Olshen, R. A., and Stone, C. J. Classification and Regression Trees. *Biometrics*, 40(3):874, September 1984. ISSN 0006341X. doi: 10.2307/2530946. URL https://www.jstor.org/stable/2530946?origin=crossref.

Breslow, N. E. and Day, N. E. Statistical methods in cancer research. Volume I - The analysis of case-control studies. *IARC scientific publications*, (32):5–338, 1980. ISSN 0300-5038.

Breunig, M. M., Kriegel, H.-P., Ng, R. T., and Sander, J. LOF: identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, pp. 93–104, Dallas Texas USA, May 2000. ACM. ISBN 978-1-58113-217-5. doi: 10.1145/342009.335388. URL https://dl.acm.org/doi/10.1145/342009.335388.

Brunton, S. L., Proctor, J. L., and Kutz, J. N. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences*, 113(15):3932–3937, March 2016. doi: 10.1073/pnas.1517384113. URL https://doi.org/10.1073%2Fpnas.1517384113. Publisher: Proceedings of the National Academy of Sciences.

Buntine, W. Theory refinement on Bayesian networks. In *Uncertainty proceedings 1991*, pp. 52–60. Elsevier, 1991. URL https://www.sciencedirect.com/science/article/pii/B9781558602038500103.

Burkart, N. and Huber, M. F. A survey on the explainability of supervised machine learning. *Journal of Artificial Intelligence Research*, 70:245–317, 2021. URL https://www.jair.org/index.php/jair/article/view/12228.

Bénard, C., Biau, G., Da Veiga, S., and Scornet, E. SIRUS: Stable and Interpretable RUle Set for classification. *Electronic Journal of Statistics*, 15:427–505, January 2021. doi: 10.1214/20-EJS1792.

Camps-Valls, G., Gerhardus, A., Ninad, U., Varando, G., Martius, G., Balaguer-Ballester, E., Vinuesa, R., Diaz, E., Zanna, L., and Runge, J. Discovering Causal Relations and Equations from Data, May 2023. URL http://arxiv.org/abs/2305.13341. Issue: arXiv:2305.13341 arXiv:2305.13341 [physics, stat].

Capistrán, C., Constandse, C., and Ramos-Francia, M. Multi-horizon inflation forecasts using disaggregated data. *Economic Modelling*, 27(3):666–677, 2010. Publisher: Elsevier.

Caruana, R., Lou, Y., Gehrke, J., Koch, P., Sturm, M., and Elhadad, N. Intelligible Models for HealthCare: Predicting Pneumonia Risk and Hospital 30-day Readmission. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1721–1730, Sydney NSW Australia, August 2015. ACM. ISBN 978-1-4503-3664-2. doi: 10.1145/2783258.2788613. URL https://dl.acm.org/doi/10.1145/2783258.2788613.

Carvalho, D. V., Pereira, E. M., and Cardoso, J. S. Machine Learning Interpretability: A Survey on Methods and Metrics. *Electronics*, 8(8):832, August 2019. ISSN 2079-9292. doi: 10.3390/electronics8080832. URL https://www.mdpi.com/2079-9292/8/8/832. Number: 8 Publisher: Multidisciplinary Digital Publishing Institute.

Chaddad, A., Peng, J., Xu, J., and Bouridane, A. Survey of Explainable AI Techniques in Healthcare. *Sensors*, 23(2):634, January 2023. ISSN 1424-8220. doi: 10.3390/s23020634. URL https://www.mdpi.com/1424-8220/23/2/634. Number: 2 Publisher: Multidisciplinary Digital Publishing Institute.

Champion, K., Zheng, P., Aravkin, A. Y., Brunton, S. L., and Kutz, J. N. A unified sparse optimization framework to learn parsimonious physics-informed models from data, July 2020. URL http://arxiv.org/abs/1906.10612. Issue: arXiv:1906.10612 arXiv:1906.10612 [physics].

Chan, H. and Darwiche, A. On the Robustness of Most Probable Explanations, June 2012. URL http://arxiv.org/abs/1206.6819. arXiv:1206.6819 [cs].

Chen, T., Shang, C., Su, P., and Shen, Q. Induction of accurate and interpretable fuzzy rules from preliminary crisp representation. *Knowledge-Based Systems*, 146:152–166, April 2018. ISSN 0950-7051. doi: 10.1016/j.knosys.2018.02.003. URL https://www.sciencedirect.com/science/article/pii/S0950705118300546.

Chen, Y., Rangarajan, G., Feng, J., and Ding, M. Analyzing multiple nonlinear time series with extended Granger causality. *Physics Letters A*, 324(1):26 – 35, 2004. ISSN 0375-9601. doi: https://doi.org/10.1016/j.physleta.2004.02.032. URL http://www.sciencedirect.com/science/article/pii/S0375960104002403.

Chickering, D. M. Learning Bayesian Networks is NP-Complete. In Bickel, P., Diggle, P., Fienberg, S., Krickeberg, K., Olkin, I., Wermuth, N., Zeger, S., Fisher, D., and Lenz, H.-J. (eds.), *Learning from Data*, volume 112, pp. 121–130. Springer New York, New York, NY, 1996. ISBN 978-0-387-94736-5 978-1-4612-2404-4. doi: 10.1007/978-1-4612-2404-4_12. URL http://link.springer.com/10.1007/978-1-4612-2404-4_12. Series Title: Lecture Notes in Statistics.

Choi, E., Bahadori, M. T., Sun, J., Kulas, J., Schuetz, A., and Stewart, W. Retain: An interpretable predictive model for healthcare using reverse time attention mechanism. *Advances in neural information processing systems*, 29, 2016.

Cooper, G. F. and Herskovits, E. A Bayesian method for the induction of probabilistic networks from data. *Machine Learning*, 9(4):309–347, October 1992. ISSN 0885-6125, 1573-0565. doi: 10.1007/BF00994110. URL http://link.springer.com/10.1007/BF00994110.

Dabrowski, J. J., Zhang, Y., and Rahman, A. ForecastNet: a time-variant deep feed-forward neural network architecture for multi-step-ahead time-series forecasting. In *International conference on neural information processing*, pp. 579–591. Springer, 2020.

De Gooijer, J. G. and Hyndman, R. J. 25 years of time series forecasting. *International journal of forecasting*, 22(3):443–473, 2006. URL https://www.sciencedirect.com/science/article/pii/S0169207006000021. Publisher: Elsevier.

De Stefani, J. *Towards multivariate multi-step-ahead time series forecasting: A machine learning perspective*. PhD Thesis, Université de Mons, 2022. URL https://difusion.ulb.ac.be/vufind/Record/ULB-DIPOT:oai:dipot.ulb.ac.be:2013/340052/Details. Publisher: Université libre de Bruxelles.

Degas, A., Islam, M. R., Hurter, C., Barua, S., Rahman, H., Poudel, M., Ruscio, D., Ahmed, M. U., Begum, S., Rahman, M. A., Bonelli, S., Cartocci, G., Di Flumeri, G., Borghini, G., Babiloni, F., and Aricó, P. A Survey on Artificial Intelligence (AI) and eXplainable AI in Air Traffic Management: Current Trends and Development with Future Research Trajectory. *Applied Sciences*, 12(3):1295, January 2022. ISSN 2076-3417. doi: 10.3390/app12031295. URL https://www.mdpi.com/2076-3417/12/3/1295. Number: 3 Publisher: Multidisciplinary Digital Publishing Institute.

Dhaou, A., Bertoncello, A., Gourvénec, S., Garnier, J., and Le Pennec, E. Causal and Interpretable Rules for Time Series Analysis. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pp. 2764–2772, Virtual Event Singapore, August 2021. ACM. ISBN 978-1-4503-8332-5. doi: 10.1145/3447548.3467161. URL https://dl.acm.org/doi/10.1145/3447548.3467161.

Donoho, D. L. Compressed sensing. *IEEE Transactions on information theory*, 52(4):1289–1306, 2006. URL https://ieeexplore.ieee.org/abstract/document/1614066/. Publisher: IEEE.

Doran, D., Schulz, S., and Besold, T. R. What Does Explainable AI Really Mean? A New Conceptualization of Perspectives, October 2017. URL http://arxiv.org/abs/1710.00794. arXiv:1710.00794 [cs].

Doshi-Velez, F. and Kim, B. Towards A Rigorous Science of Interpretable Machine Learning, March 2017. URL http://arxiv.org/abs/1702.08608. arXiv:1702.08608 [cs, stat].

Du, M., Liu, N., and Hu, X. Techniques for interpretable machine learning. *Communications of the ACM*, 63(1):68–77, December 2019. ISSN 0001-0782, 1557-7317. doi: 10.1145/3359786. URL https://dl.acm.org/doi/10.1145/3359786.

Dubois, D. and Prade, H. What are fuzzy rules and how to use them. *Fuzzy Sets and Systems*, 84, dedicated to the Memory of Professor Arnold Kaufmann(2):169–185, December 1996. doi: 10.1016/0165-0114(96)00066-8. URL https://hal.science/hal-04039815. Publisher: Elsevier.

Duncker, L., Bohner, G., Boussard, J., and Sahani, M. Learning interpretable continuous-time models of latent stochastic dynamical systems. In *Proceedings of the 36th International Conference on Machine Learning*, pp. 1726–1734. PMLR, May 2019. URL https://proceedings.mlr.press/v97/duncker19a.html. ISSN: 2640-3498.

Dupré, A., Drobinski, P., Alonzo, B., Badosa, J., Briard, C., and Plougonven, R. Sub-hourly forecasting of wind speed and wind energy. *Renewable Energy*, 145:2373–2379, 2020a. URL https://www.sciencedirect.com/science/article/pii/S0960148119311814. Publisher: Elsevier.

Dupré, A., Drobinski, P., Badosa, J., Briard, C., and Tankov, P. The economic value of wind energy nowcasting. *Energies*, 13(20):5266, 2020b. URL https://www.mdpi.com/1996-1073/13/20/5266. Publisher: MDPI.

Džeroski, S. and Todorovski, L. Discovering dynamics. In *Proc. tenth international conference on machine learning*, pp. 97–103, 1993. URL https://books.google.nl/books?hl=en&lr=&id=TrqjBQAAQBAJ&oi=fnd&pg=PA97&dq=Sa%C5%A1o+D%C5%BEeroski+and+Ljup%C3%A9o+Todorovski.+Discovering+dynamics.+In+Proc.+tenth+international+conference+on+machine+learning,+pp.+97%E2%80%93103,+1993.&ots=v4S7PP7S_l&sig=vAx3Zm5MVAFutah5OUyjty3t0Q0.

Entner, D. and Hoyer, P. On Causal Discovery from Time Series Data using FCI. *Proceedings of the 5th European Workshop on Probabilistic Graphical Models, PGM 2010*, September 2010.

Estberg, L., Gardner, I. A., Stover, S. M., and Johnson, B. J. A case-crossover study of intensive racing and training schedules and risk of catastrophic musculoskeletal injury and lay-up in California Thoroughbred racehorses. *Preventive Veterinary Medicine*, 33(1-4): 159–170, 1998. URL https://www.sciencedirect.com/science/article/pii/S0167587797000470. Publisher: Elsevier.

European Centre for Medium-Range Weather Forecast. ECMWF, March 2024. URL https://www.ecmwf.int/.

{European Commission}. Loi sur l'intelligence artificielle de l'UE, 2021. URL https://artificialintelligenceact.eu/fr/l-acte/.

{European Parliament}. Artificial Intelligence Act: deal on comprehensive rules for trustworthy AI | News | European Parliament, December 2023. URL https://www.europarl.europa.eu/news/en/press-room/20231206IPR15699/artificial-intelligence-act-deal-on-comprehensive-rules-for-trustworthy-ai.

Faes, L., Nollo, G., and Chon, K. H. Assessment of Granger Causality by Nonlinear Model Identification: Application to Short-term Cardiovascular Variability. *Annals of Biomedical Engineering*, 36(3):381–395, March 2008. ISSN 0090-6964, 1573-9686. doi: 10.1007/s10439-008-9441-z. URL http://link.springer.com/10.1007/s10439-008-9441-z.

Falcon, A. Aristotle on Causality. In Zalta, E. N. and Nodelman, U. (eds.), *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, spring 2023 edition, 2023. URL https://plato.stanford.edu/archives/spr2023/entries/aristotle-causality/.

Ferreira, C. *Gene expression programming: mathematical modeling by an artificial intelligence*, volume 21. Springer, 2006. URL https://books.google.nl/books?hl=en&lr=&id=NkG7BQAAQBAJ&oi=fnd&pg=PR7&dq=Gene+expression+programming:+mathematical+modeling+by+an+artificial+intelligence&ots=Y-jnsyZhC2&sig=dD9sXpOWg_HLy2fbMllRI8rq3Nk.

Fournier-Viger, P., Lin, J. C., Vo, B., Chi, T. T., Zhang, J., and Le, H. B. A survey of itemset mining. *WIREs Data Mining and Knowledge Discovery*, 7(4):e1207, July 2017. ISSN 1942-4787, 1942-4795. doi: 10.1002/widm.1207. URL https://wires.onlinelibrary.wiley.com/doi/10.1002/widm.1207.

{France AI Strategy Report}. France AI Strategy Report - European Commission. Technical report, 2018. URL https://ai-watch.ec.europa.eu/countries/france/france-ai-strategy-report_en.

Friedman, J. H. and Popescu, B. E. Predictive learning via rule ensembles. *The Annals of Applied Statistics*, 2(3), September 2008. ISSN 1932-6157. doi: 10.1214/07-AOAS148. URL http://arxiv.org/abs/0811.1679. arXiv:0811.1679 [stat].

Friedman, N., Nachman, I., and Pe'er, D. Learning Bayesian Network Structure from Massive Datasets: The "Sparse Candidate" Algorithm, January 2013. URL http://arxiv.org/abs/1301.6696. arXiv:1301.6696 [cs, stat].

{Future of Life Institute}. Policymaking In The Pause - Future of Life Institute, 2023. URL https://futureoflife.org/document/policymaking-in-the-pause/.

Gasparrini, A. and Leone, M. Attributable risk from distributed lag models. *BMC Medical Research Methodology*, 14(1):55, December 2014. ISSN 1471-2288. doi: 10.1186/1471-2288-14-55. URL https://bmcmedresmethodol.biomedcentral.com/articles/10.1186/1471-2288-14-55.

Gasparrini, A., Armstrong, B., and Kenward, M. G. Distributed lag non-linear models. *Statistics in Medicine*, 29(21):2224–2234, September 2010. ISSN 0277-6715, 1097-0258. doi: 10.1002/sim.3940. URL https://onlinelibrary.wiley.com/doi/10.1002/sim.3940.

Geweke, J. Measurement of Linear Dependence and Feedback between Multiple Time Series. *Journal of the American Statistical Association*, 77(378):304–313, June 1982. ISSN 0162-1459, 1537-274X. doi: 10.1080/01621459.1982.10477803. URL http://www.tandfonline.com/doi/abs/10.1080/01621459.1982.10477803.

Gilpin, L. H., Bau, D., Yuan, B. Z., Bajwa, A., Specter, M., and Kagal, L. Explaining explanations: An overview of interpretability of machine learning. In *2018 IEEE 5th*

International Conference on data science and advanced analytics (DSAA), pp. 80–89. IEEE, 2018. URL https://ieeexplore.ieee.org/abstract/document/8631448.

Glennan, S. Mechanisms and the nature of causation. *Erkenntnis*, 44(1), January 1996. ISSN 0165-0106, 1572-8420. doi: 10.1007/BF00172853. URL http://link.springer.com/10.1007/BF00172853.

Glymour, C., Zhang, K., and Spirtes, P. Review of causal discovery methods based on graphical models. *Frontiers in genetics*, 10:524, 2019. URL https://www.frontiersin.org/articles/10.3389/fgene.2019.00524/full. Publisher: Frontiers Media SA.

Gong, M., Zhang, K., Schoelkopf, B., Tao, D., and Geiger, P. Discovering temporal causal relations from subsampled data. In *International Conference on Machine Learning*, pp. 1898–1906. PMLR, 2015. URL http://proceedings.mlr.press/v37/gongb15.html.

Granger, C. W. J. Investigating Causal Relations by Econometric Models and Cross-spectral Methods. *Econometrica*, 37(3):424–438, 1969. ISSN 00129682, 14680262. URL http://www.jstor.org/stable/1912791. Publisher: [Wiley, Econometric Society].

Greenland, S. and Robins, J. M. Identifiability, Exchangeability, and Epidemiological Confounding. *International Journal of Epidemiology*, 15(3):413–419, 1986. ISSN 0300-5771, 1464-3685. doi: 10.1093/ije/15.3.413. URL https://academic.oup.com/ije/article-lookup/doi/10.1093/ije/15.3.413.

Griffin, R. P., Tatar, U., and Yankson, B. *ICCWS 2022 17th International Conference on Cyber Warfare and Security*. Academic Conferences and Publishing Limited, March 2022. ISBN 978-1-914587-27-6. Google-Books-ID: Shd2EAAAQBAJ.

Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., and Pedreschi, D. A Survey of Methods for Explaining Black Box Models. *ACM Computing Surveys*, 51 (5):1–42, September 2019. ISSN 0360-0300, 1557-7341. doi: 10.1145/3236009. URL https://dl.acm.org/doi/10.1145/3236009.

Guimerà, R., Reichardt, I., Aguilar-Mogas, A., Massucci, F. A., Miranda, M., Pallarès, J., and Sales-Pardo, M. A Bayesian machine scientist to aid in the solution of challenging scientific problems. *Science Advances*, 6(5):eaav6971, January 2020. ISSN 2375-2548. doi: 10.1126/sciadv.aav6971. URL https://www.science.org/doi/10.1126/sciadv.aav6971.

Gunning, D. Explainable artificial intelligence (xai). *Defense advanced research projects agency (DARPA), nd Web*, 2(2):1, 2017. URL https://nsarchive.gwu.edu/sites/default/files/documents/5794867/National-Security-Archive-David-Gunning-DARPA.pdf.

Gunning, D. and Aha, D. DARPA's Explainable Artificial Intelligence (XAI) Program. *AI Magazine*, 40(2):44–58, June 2019. ISSN 2371-9621. doi: 10.1609/aimag.v40i2.2850. URL https://ojs.aaai.org/aimagazine/index.php/aimagazine/article/view/2850. Number: 2.

Hadri, S., Najib, M., Bakhouya, M., Fakhri, Y., and El Arroussi, M. Performance Evaluation of Forecasting Strategies for Electricity Consumption in Buildings. *Energies*, 14(18):5831, 2021. Publisher: MDPI.

Halpern, J. Y. and Pearl, J. Causes and Explanations: A Structural-Model Approach. Part II: Explanations. *The British Journal for the Philosophy of Science*, 56(4):889–911, December 2005. ISSN 0007-0882, 1464-3537. doi: 10.1093/bjps/axi148. URL https://www.journals.uchicago.edu/doi/10.1093/bjps/axi148.

Hamilton, J. D. *Time series analysis*. Princeton University Press, Princeton, N.J, 1994. ISBN 978-0-691-04289-3.

Han, J., Pei, J., Yin, Y., and Mao, R. Mining Frequent Patterns without Candidate Generation: A Frequent-Pattern Tree Approach. *Data Mining and Knowledge Discovery*, 8(1):53–87, January 2004. ISSN 1573-756X. doi: 10.1023/B:DAMI.0000005258.31418.83. URL https://doi.org/10.1023/B:DAMI.0000005258.31418.83.

Hastie, T. and Zou, H. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 67(2):301–320, 2005. URL https://academic.oup.com/jrsssb/article-abstract/67/2/301/7109482. Publisher: Oxford University Press.

Hastie, T., Tibshirani, R., and Friedman, J. The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2nd Edition. In *Springer Series in Statistics*, 2009.

He, C., Ma, M., and Wang, P. Extract interpretability-accuracy balanced rules from artificial neural networks: A review. *Neurocomputing*, 387:346–358, 2020. URL https://www.sciencedirect.com/science/article/pii/S0925231220300801. Publisher: Elsevier.

Hebert, C., Delaney, J. A. C., Hemmelgarn, B., Lévesque, L. E., and Suissa, S. Benzodiazepines and elderly drivers: a comparison of pharmacoepidemiological study designs. *Pharmacoepidemiology and Drug Safety*, 16(8):845–849, August 2007. ISSN 1053-8569, 1099-1557. doi: 10.1002/pds.1432. URL https://onlinelibrary.wiley.com/doi/10.1002/pds.1432.

Heckerman, D., Geiger, D., and Chickering, D. M. Learning Bayesian networks: The combination of knowledge and statistical data. *Machine Learning*, 20(3):197–243, September 1995. ISSN 0885-6125, 1573-0565. doi: 10.1007/BF00994016. URL http://link.springer.com/10.1007/BF00994016.

Hempel, C. G. and Oppenheim, P. Studies in the Logic of Explanation. *Philosophy of science*, 15(2):135–175, 1948. URL https://www.cambridge.org/core/journals/philosophy-of-science/article/studies-in-the-logic-of-explanation/411C0354383994902A993DD7A08DBBC4. Publisher: Cambridge University Press.

Herlocker, J. L., Konstan, J. A., and Riedl, J. Explaining collaborative filtering recommendations. In *Proceedings of the 2000 ACM conference on Computer supported cooperative work*, pp. 241–250, Philadelphia Pennsylvania USA, December 2000. ACM. ISBN 978-1-58113-222-9. doi: 10.1145/358916.358995. URL https://dl.acm.org/doi/10.1145/358916.358995.

Hinton, G. E. and Roweis, S. Stochastic Neighbor Embedding. In *Advances in Neural Information Processing Systems*, volume 15. MIT Press, 2002. URL https://proceedings.neurips.cc/paper_files/paper/2002/hash/6150ccc6069bea6b5716254057a194ef-Abstract.html.

Hofeditz, L., Mirbabaie, M., Luther, A., Mauth, R., and Rentemeister, I. Ethics Guidelines for Using AI-based Algorithms in Recruiting: Learnings from a Systematic Literature Review. 2022. doi: 10.24251/HICSS.2022.018. URL http://hdl.handle.net/10125/79348.

Holland, J. H. *Induction: Processes of inference, learning, and discovery*. MIT press, 1986. URL https://books.google.nl/books?hl=en&lr=&id=Z6EFBaLApE8C&oi=fnd&pg=PR13&dq=Holland,+John%3B+Holyoak,+Keith%3B+Nisbett,+Richard%3B+Thagard,+Paul+(1986)+Induction:+Processes+of+Inference,+Learning,+and+Discovery.+Cambridge:+MIT+Press&ots=2sTvhqKPGJ&sig=OCbLz3gKy5zYZyS1y72OUghE41w.

Hoyer, P., Janzing, D., Mooij, J. M., Peters, J., and Schölkopf, B. Nonlinear causal discovery with additive noise models. *Advances in neural information processing systems*, 21, 2008. URL https://proceedings.neurips.cc/paper/2008/hash/f7664060cc52bc6f3d620bcedc94a4b6-Abstract.html.

Huang, J., Jiao, Y., Liu, Y., and Lu, X. A constructive approach to L_0 penalized regression. *Journal of Machine Learning Research*, 19(10):1–37, 2018. URL https://www.jmlr.org/papers/v19/17-194.html.

Hume, D. *An Enquiry Concerning the Human Understanding: And An Enquiry Concerning the Principles of Morals*. Clarendon Press, 1894. Google-Books-ID: DTU7AAAAYAAJ.

Hyndman, R. J. and Athanasopoulos, G. *Forecasting: principles and practice*. OTexts, 2018. URL https://books.google.nl/books?hl=en&lr=&id=_bBhDwAAQBAJ&oi=fnd&pg=PA7&dq=hyndman+forecasting&ots=Tje2zgWRMH&sig=0EXPTPvATx0tQNOeBwt9cC_IUWo.

Hyvärinen, A., Zhang, K., Shimizu, S., and Hoyer, P. O. Estimation of a structural vector autoregression model using non-gaussianity. *Journal of Machine Learning Research*, 11(5), 2010. URL https://www.jmlr.org/papers/volume11/hyvarinen10a/hyvarinen10a.pdf.

Imbens, G. W. and Rubin, D. B. *Causal Inference for Statistics, Social, and Biomedical Sciences: An Introduction*. Cambridge University Press, Cambridge, 2015. ISBN 978-0-521-88588-1. doi: 10.1017/CBO9781139025751. URL https://www.cambridge.org/core/books/causal-inference-for-statistics-social-and-biomedical-sciences/71126BE90C58F1A431FE9B2DD07938AB.

Ishikawa, K. and Ishikawa, K. *Guide to quality control*, volume 2. Asian Productivity Organization Tokyo, 1982. URL https://pascal-francis.inist.fr/vibad/index.php?action=getRecordDetail&idt=6404027.

Jaakkola, J. J. K. Case-crossover design in air pollution epidemiology. *European Respiratory Journal*, 21(40 suppl):81s–85s, 2003. URL https://erj.ersjournals.com/content/21/40_suppl/81s.short. Publisher: Eur Respiratory Soc.

Januschowski, T., Gasthaus, J., Wang, Y., Salinas, D., Flunkert, V., Bohlke-Schneider, M., and Callot, L. Criteria for Classifying Forecasting Methods, December 2022. URL http://arxiv.org/abs/2212.03523. arXiv:2212.03523 [cs, stat].

Jeschke, J., Sun, D., Jamshidnejad, A., and De Schutter, B. Grammatical-Evolution-based parameterized Model Predictive Control for urban traffic networks. *Control Engineering Practice*, 132:105431, 2023. URL https://www.sciencedirect.com/science/article/pii/S0967066122002623. Publisher: Elsevier.

Josephson, J. R. and Josephson, S. G. *Abductive inference: Computation, philosophy, technology*. Cambridge University Press, 1996. URL https://books.google.nl/books?hl=en&lr=&id=uu6zXrogwWAC&oi=fnd&pg=PA1&dq=J.+R.+Josephson,+S.+G.+Josephson,+Abductive+inference:+Computation,+philosophy,+technology,+Cambridge+University+Press,+1996&ots=6QqvO4S1uz&sig=EutmO1uFpz_oDzW9hsj4B33FgXg.

Judge, G. G., Griffiths, W. E., Hill, R. C., Lütkepohl, H., and Lee, T.-C. *The theory and practice of econometrics*, volume 49. John wiley & sons, 1991. URL https://books.google.nl/books?hl=en&lr=&id=tJfEEAAAQBAJ&oi=fnd&pg=PA1020&dq=the+theory+and+practice+of+econometric&ots=Fmrwrh53zz&sig=26B2kwajk7R9mZFyaHjWr1145lw.

kdnuggets. Most Popular Distance Metrics Used in KNN and When to Use Them. URL https://www.kdnuggets.com/most-popular-distance-metrics-used-in-knn-and-when-to-use-them. Section: 2020 Nov Tutorials, Overviews.

Kim, B., Khanna, R., and Koyejo, O. O. Examples are not enough, learn to criticize! Criticism for Interpretability. In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper/2016/hash/5680522b8e2bb01943234bce7bf84534-Abstract.html.

KISTER, H. Z; HAAS, J. R. Predict entrainment flooding on sieve and valve trays. *Chemical engineering progress*, 1990. ISSN 0360-7275.

Kitson, N. K., Constantinou, A. C., Guo, Z., Liu, Y., and Chobtham, K. A survey of Bayesian Network structure learning. *Artificial Intelligence Review*, 56(8):8721–8814, August 2023. ISSN 1573-7462. doi: 10.1007/s10462-022-10351-w. URL https://doi.org/10.1007/s10462-022-10351-w.

Koza, J. Genetic programming as a means for programming computers by natural selection. *Statistics and Computing*, 4(2), June 1994. ISSN 0960-3174, 1573-1375. doi: 10.1007/BF00175355. URL http://link.springer.com/10.1007/BF00175355.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. URL https://papers.nips.cc/paper_files/paper/2012/hash/c399862d3b9d6b76c8436e924a68c45b-Abstract.html.

Kuo, C.-L., Duan, Y., and Grady, J. Unconditional or conditional logistic regression model for age-matched case–control data? *Frontiers in public health*, 6:57, 2018. URL https://www.frontiersin.org/articles/10.3389/fpubh.2018.00057/full. Publisher: Frontiers Media SA.

La Cava, W., Orzechowski, P., Burlacu, B., de Franca, F., Virgolin, M., Jin, Y., Kommenda, M., and Moore, J. Contemporary Symbolic Regression Methods and their Relative Performance. *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, 1, December 2021. URL https://datasets-benchmarks-proceedings.neurips.cc/paper/2021/hash/c0c7c76d30bd3dcaefc96f40275bdc0a-Abstract-round1.html.

Lakkaraju, H., Kamar, E., Caruana, R., and Leskovec, J. Interpretable & Explorable Approximations of Black Box Models, July 2017. URL http://arxiv.org/abs/1707.01154. arXiv:1707.01154 [cs].

Laurent, O., Bard, D., Filleul, L., and Segala, C. Effect of socioeconomic status on the relationship between atmospheric pollution and mortality. *Journal of epidemiology and community health*, 61(8):665, 2007. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2652988/. Publisher: BMJ Publishing Group.

Lee, J. T. and Schwartz, J. Reanalysis of the effects of air pollution on daily mortality in Seoul, Korea: A case-crossover design. *Environmental Health Perspectives*, 107(8): 633–636, August 1999. ISSN 0091-6765, 1552-9924. doi: 10.1289/ehp.99107633. URL https://ehp.niehs.nih.gov/doi/10.1289/ehp.99107633.

Lenca, P., Vaillant, B., Meyer, P., and Lallich, S. Association Rule Interestingness Measures: Experimental and Theoretical Studies. In *Quality Measures in Data Mining*, volume 43, pp. 51–76. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007. ISBN 978-3-540-44911-9 978-3-540-44918-8. doi: 10.1007/978-3-540-44918-8_3. URL http://link.springer.com/10.1007/978-3-540-44918-8_3. Series Title: Studies in Computational Intelligence.

Letham, B., Rudin, C., McCormick, T. H., and Madigan, D. Interpretable classifiers using rules and Bayesian analysis: Building a better stroke prediction model. *The Annals of Applied Statistics*, 9(3), September 2015. ISSN 1932-6157. doi: 10.1214/15-AOAS848. URL http://arxiv.org/abs/1511.01644. arXiv:1511.01644 [cs, stat].

Lewis, D. Counterfactuals and Comparative Possibility. In Harper, W. L., Stalnaker, R., and Pearce, G. (eds.), *IFS*, pp. 57–85. Springer Netherlands, Dordrecht, 1973. ISBN 978-90-277-1220-2 978-94-009-9117-0. doi: 10.1007/978-94-009-9117-0_3. URL http://link.springer.com/10.1007/978-94-009-9117-0_3.

Lewis, D. Causal explanation. 1986. URL https://philpapers.org/rec/LEWCE.

Lewis, R. J. An introduction to classification and regression tree (CART) analysis. In *Annual meeting of the society for academic emergency medicine in San Francisco, California*, volume 14. Department of Emergency Medicine Harbor-UCLA Medical Center Torrance San ..., 2000. URL https://www.researchgate.net/profile/Roger-Lewis-8/publication/240719582_An_Introduction_to_Classification_and_Regression_Tree_CART_Analysis/links/0046352d3fb18f1740000000/An-Introduction-to-Classification-and-Regression-Tree-CART-Analysis.pdf.

Li, S., Jin, X., Xuan, Y., Zhou, X., Chen, W., Wang, Y.-X., and Yan, X. Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting. *Advances in neural information processing systems*, 32, 2019.

Liao, Q. V. and Varshney, K. R. Human-Centered Explainable AI (XAI): From Algorithms to User Experiences, April 2022. URL http://arxiv.org/abs/2110.10790. arXiv:2110.10790 [cs].

Liao, Z. A., Sharma, C., Cussens, J., and van Beek, P. Finding all Bayesian network structures within a factor of optimal. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pp. 7892–7899, 2019. URL https://aaai.org/ojs/index.php/AAAI/article/view/4788. Issue: 01.

Lichtarowicz, M. Distillation, 2016. URL https://www.essentialchemicalindustry.org/processes/distillation.html.

Lim, B. Forecasting treatment responses over time using recurrent marginal structural networks. *advances in neural information processing systems*, 31, 2018.

Lim, B., Arık, S. O., Loeff, N., and Pfister, T. Temporal Fusion Transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting*, 37(4):1748–1764, October 2021. ISSN 0169-2070. doi: 10.1016/j.ijforecast.2021.03.012. URL https://www.sciencedirect.com/science/article/pii/S0169207021000637.

Lin, C.-W., Hong, T.-P., and Lu, W.-H. A two-phase fuzzy mining approach. In *International conference on fuzzy systems*, pp. 1–5. IEEE, 2010. URL https://ieeexplore.ieee.org/abstract/document/5584373/.

Lin, J., Keogh, E., Wei, L., and Lonardi, S. Experiencing SAX: a novel symbolic representation of time series. *Data Mining and Knowledge Discovery*, 15(2):107–144, August 2007. ISSN 1384-5810, 1573-756X. doi: 10.1007/s10618-007-0064-z. URL http://link.springer.com/10.1007/s10618-007-0064-z.

Linardatos, P., Papastefanopoulos, V., and Kotsiantis, S. Explainable AI: A Review of Machine Learning Interpretability Methods. *Entropy*, 23(1):18, January 2021. ISSN 1099-4300. doi: 10.3390/e23010018. URL https://www.mdpi.com/1099-4300/23/1/18. Number: 1 Publisher: Multidisciplinary Digital Publishing Institute.

Lipton, Z. C. The Mythos of Model Interpretability, March 2017. URL http://arxiv.org/abs/1606.03490. arXiv:1606.03490 [cs, stat].

Liu, F. T., Ting, K. M., and Zhou, Z.-H. Isolation forest. In *2008 eighth ieee international conference on data mining*, pp. 413–422. IEEE, 2008. URL https://ieeexplore.ieee.org/abstract/document/4781136/.

Liu, H., Chen, C., Lv, X., Wu, X., and Liu, M. Deterministic wind energy forecasting: A review of intelligent predictors and auxiliary methods. *Energy Conversion and Management*, 195:328–345, 2019. Publisher: Elsevier.

Lokrantz, A., Gustavsson, E., and Jirstrand, M. Root cause analysis of failures and quality deviations in manufacturing using machine learning. *Procedia Cirp*, 72:1057–1062, 2018. URL https://www.sciencedirect.com/science/article/pii/S2212827118303895. Publisher: Elsevier.

Lopes, P., Silva, E., Braga, C., Oliveira, T., and Rosado, L. XAI Systems Evaluation: A Review of Human and Computer-Centred Methods. *Applied Sciences*, 12(19):9423, January 2022. ISSN 2076-3417. doi: 10.3390/app12199423. URL https://www.mdpi.com/2076-3417/12/19/9423. Number: 19 Publisher: Multidisciplinary Digital Publishing Institute.

Ludwig, O., Nunes, U., Araújo, R., Schnitman, L., and Lepikson, H. Applications of information theory, genetic algorithms, and neural models to predict oil flow. *Communications in Nonlinear Science and Numerical Simulation*, 14:2870–2885, 2009.

Lundberg, S. M. and Lee, S.-I. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/8a20a8621978632d76c43dfd28b67767-Abstract.html.

Lütkepohl, H. Vector autoregressive models. *Handbook of research methods and applications in empirical macroeconomics*, 30, 2013. URL https://books.google.nl/books?hl=en&lr=&id=ff4BAQAAQBAJ&oi=fnd&pg=PA139&dq=Lutkepohl,+2013&ots=7cAWe0sbDh&sig=ASZzNkt_Enp0g8YnMCbi9XLZjJw. Publisher: Edward Elgar Publishing Cheltenham, UK.

Maclure, M. The Case-Crossover Design: A Method for Studying Transient Effects on the Risk of Acute Events. *American journal of epidemiology*, 133:144–153, 1991. doi: 10.1093/oxfordjournals.aje.a115853.

Maclure, M. Triggering of acute myocardial infarction by physical exertion: protection against triggering by regular exertion. *N. Engl. J. Med*, 329:1677–1683, 1993.

Maclure, M. Triggering of acute myocardial infarction by episodes of anger. *Circulation*, 92: 1720–1725, 1995.

Maclure, M. and Mittleman, A. M. A. Should We Use a Case-Crossover Design? *Annual Review of Public Health*, 21(1):193–221, May 2000. ISSN 0163-7525, 1545-2093. doi: 10.1146/annurev.publhealth.21.1.193. URL https://www.annualreviews.org/doi/10.1146/annurev.publhealth.21.1.193.

Maclure, M. and Mittleman, M. A. Cautions about Car Telephones and Collisions. *New England Journal of Medicine*, 336(7):501–502, February 1997. ISSN 0028-4793, 1533-4406. doi: 10.1056/NEJM199702133360709. URL http://www.nejm.org/doi/abs/10.1056/NEJM199702133360709.

Mahmood, M. S. Fit Non-Linear Relationship Using Generalized Additive Model, November 2021. URL https://towardsdatascience.com/fit-non-linear-relationship-using-generalized-additive-model-53a334201b5d.

Makke, N. and Chawla, S. Interpretable scientific discovery with symbolic regression: a review. *Artificial Intelligence Review*, 57(1):2, January 2024. ISSN 0269-2821, 1573-7462. doi: 10.1007/s10462-023-10622-0. URL https://link.springer.com/10.1007/s10462-023-10622-0.

Mamdani, E. H. and Assilian, S. An experiment in linguistic synthesis with a fuzzy logic controller. *International journal of man-machine studies*, 7(1):1–13, 1975. URL https://www.sciencedirect.com/science/article/pii/S0020737375800022. Publisher: Elsevier.

Marcinkevičs, R. and Vogt, J. E. Interpretable and explainable machine learning: A methods-centric overview with concrete examples. *WIREs Data Mining and Knowledge Discovery*, 13(3):e1493, 2023. ISSN 1942-4795. doi: 10.1002/widm.1493. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/widm.1493. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/widm.1493.

Margaritis, D. and Thrun, S. Bayesian Network Induction via Local Neighborhoods. In *Advances in Neural Information Processing Systems*, volume 12. MIT Press, 1999. URL https://papers.nips.cc/paper_files/paper/1999/hash/5d79099fcdf499f12b79770834c0164a-Abstract.html.

Marinazzo D., Pellicoro M., S. S. Kernel method for nonlinear Granger causality. *Phys. Rev. Lett.*, 100:144103, 2008.

Markus, A. F., Kors, J. A., and Rijnbeek, P. R. The role of explainability in creating trustworthy artificial intelligence for health care: A comprehensive survey of the terminology, design choices, and evaluation strategies. *Journal of Biomedical Informatics*, 113:103655, January 2021. ISSN 1532-0464. doi: 10.1016/j.jbi.2020.103655. URL https://www.sciencedirect.com/science/article/pii/S1532046420302835.

Martinho, A., Herber, N., Kroesen, M., and Chorus, C. Ethical issues in focus by the autonomous vehicles industry. *Transport Reviews*, 41(5):556–577, September 2021. ISSN 0144-1647, 1464-5327. doi: 10.1080/01441647.2020.1862355. URL https://www.tandfonline.com/doi/full/10.1080/01441647.2020.1862355.

Mayes, G. R. Theories of explanation. 2001. URL https://philpapers.org/rec/MAYTOE.

Meier, C. R., Jick, S. S., Derby, L. E., Vasilakis, C., Jick, H., Meier, C. R., Jick, S. S., Derby, L. E., Vasilakis, C., and Jick, H. Acute respiratory-tract infections and risk of first-time acute myocardial infarction. *The Lancet*, 351(9114):1467–1471, 1998. URL https://www.thelancet.com/journals/lancet/article/PIIS0140-6736(97)11084-4/fulltext?iframe=true&width=95%EF%BF%BDght%3D95%EF%BF%BD. Publisher: Elsevier.

Meinshausen, N. Relaxed lasso. *Computational Statistics & Data Analysis*, 52(1):374–393, 2007. URL https://www.sciencedirect.com/science/article/pii/S0167947306004956. Publisher: Elsevier.

Miller, T. Explanation in Artificial Intelligence: Insights from the Social Sciences, August 2018. URL http://arxiv.org/abs/1706.07269. arXiv:1706.07269 [cs].

Mittelstadt, B., Russell, C., and Wachter, S. Explaining Explanations in AI. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pp. 279–288, Atlanta GA USA, January 2019. ACM. ISBN 978-1-4503-6125-5. doi: 10.1145/3287560.3287574. URL https://dl.acm.org/doi/10.1145/3287560.3287574.

Mittleman, M. A. and Mostofsky, E. Exchangeability in the case-crossover design. *International journal of epidemiology*, 43(5):1645–1655, 2014. URL https://academic.oup.com/ije/article-abstract/43/5/1645/695900. Publisher: Oxford University Press.

Mittleman, M. A., Maclure, M., Tofler, G. H., Sherwood, J. B., Goldberg, R. J., and Muller, J. E. Triggering of Acute Myocardial Infarction by Heavy Physical Exertion – Protection against Triggering by Regular Exertion. *New England Journal of Medicine*, 329(23):1677–1683, December 1993. ISSN 0028-4793, 1533-4406. doi: 10.1056/NEJM199312023292301. URL http://www.nejm.org/doi/abs/10.1056/NEJM199312023292301.

Mittleman, M. A., Maclure, M., Sherwood, J. B., Mulry, R. P., Tofler, G. H., Jacobs, S. C., Friedman, R., Benson, H., and Muller, J. E. Triggering of Acute Myocardial Infarction Onset by Episodes of Anger. *Circulation*, 92(7):1720–1725, October 1995. ISSN 0009-7322, 1524-4539. doi: 10.1161/01.CIR.92.7.1720. URL https://www.ahajournals.org/doi/10.1161/01.CIR.92.7.1720.

Mittleman, M. A., Maclure, M., Sherwood, J. B., Kondo, N. I., Tofler, G. H., and Muller, J. Death of a significant person increases the risk of acute MI onset. *Circulation*, 93:631, 1996.

Mittleman, M. A., Mintzer, D., Maclure, M., Tofler, G. H., Sherwood, J. B., and Muller, J. E. Triggering of Myocardial Infarction by Cocaine. *Circulation*, 99(21):2737–2741, June 1999. ISSN 0009-7322, 1524-4539. doi: 10.1161/01.CIR.99.21.2737. URL https://www.ahajournals.org/doi/10.1161/01.CIR.99.21.2737.

Molnar, C. *Interpretable machine learning*. Lulu. com, 2020. URL https://books.google.nl/books?hl=en&lr=&id=jBm3DwAAQBAJ&oi=fnd&pg=PP1&dq=molnar&ots=EhrWYpCIX3&sig=mmXLDlhUS9Nduv9HLv1Wq1Kl-tc.

Montanus, M. L. Business models for Industry 4.0: Developing a framework to determine and assess impacts on business models in the Dutch oil and gas industry. 2016. URL https://repository.tudelft.nl/islandora/object/uuid:3177d804-06d5-455c-a508-87222c1d602a/datastream/OBJ1/download.

Montavon, G., Samek, W., and Müller, K.-R. Methods for interpreting and understanding deep neural networks. *Digital signal processing*, 73:1–15, 2018. URL https://www.sciencedirect.com/science/article/pii/S1051200417302385. Publisher: Elsevier.

Moore, J. D. and Swartout, W. R. *Explanation in expert systems: A survey*. University of Southern California, Information Sciences Institute Marina del . . . , 1988. URL https://apps.dtic.mil/sti/citations/ADA206283.

Morioka, H., Hälvä, H., and Hyvarinen, A. Independent Innovation Analysis for Nonlinear Vector Autoregressive Process. In *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, pp. 1549–1557. PMLR, March 2021. URL https://proceedings.mlr.press/v130/morioka21a.html. ISSN: 2640-3498.

Morley, J., Machado, C. C. V., Burr, C., Cowls, J., Joshi, I., Taddeo, M., and Floridi, L. The ethics of AI in health care: A mapping review. *Social Science & Medicine (1982)*, 260:113172, September 2020. ISSN 1873-5347. doi: 10.1016/j.socscimed.2020.113172.

Mothilal, R. K., Sharma, A., and Tan, C. Explaining machine learning classifiers through diverse counterfactual explanations. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pp. 607–617, Barcelona Spain, January 2020. ACM. ISBN 978-1-4503-6936-7. doi: 10.1145/3351095.3372850. URL https://dl.acm.org/doi/10.1145/3351095.3372850.

Mukhiya, S. K. and Ahmed, U. *Hands-On Exploratory Data Analysis with Python: Perform EDA techniques to understand, summarize, and investigate your data*. Packt Publishing Ltd, 2020. URL https://books.google.nl/books?hl=en&lr=&id=QcHZDwAAQBAJ&oi=fnd&pg=PP1&dq=Hands-On+Exploratory+Data+Analysis+with+Python:+Perform+EDA+techniques+to+understand,+summarize,+and+investigate+your+data&ots=tQOFTnXifm&sig=_4UXg248k5lEiiHSLphmvQSilYg.

Muller, J. E., Mittleman, M. A., Maclure, M., Sherwood, J. B., and Tofler, G. H. Triggering myocardial infarction by sexual activity: low absolute risk and prevention by regular physical exertion. *Jama*, 275(18):1405–1409, 1996. URL https://jamanetwork.com/journals/jama/article-abstract/401995. Publisher: American Medical Association.

Murdoch, W. J., Singh, C., Kumbier, K., Abbasi-Asl, R., and Yu, B. Interpretable machine learning: definitions, methods, and applications. *Proceedings of the National Academy of Sciences*, 116(44):22071–22080, October 2019. ISSN 0027-8424, 1091-6490. doi: 10.1073/pnas.1900654116. URL http://arxiv.org/abs/1901.04592. arXiv:1901.04592 [cs, stat].

Nauta, M., Trienes, J., Pathak, S., Nguyen, E., Peters, M., Schmitt, Y., Schlötterer, J., Van Keulen, M., and Seifert, C. From Anecdotal Evidence to Quantitative Evaluation Methods: A Systematic Review on Evaluating Explainable AI. *ACM Computing Surveys*, 55(13s):1–42, December 2023. ISSN 0360-0300, 1557-7341. doi: 10.1145/3583558. URL https://dl.acm.org/doi/10.1145/3583558.

Neapolitan, R. E. *Learning bayesian networks*, volume 38. Pearson Prentice Hall Upper Saddle River, 2004. URL http://s2.bitdl.ir/Ebook/Computer%20Science/Artificial%20Intelligence/Learning%20Bayesian%20Networks%20-%20Neapolitan%20R.%20E..pdf.

Nelder, J. A. and Wedderburn, R. W. Generalized linear models. *Journal of the Royal Statistical Society Series A: Statistics in Society*, 135(3):370–384, 1972. URL https://academic.oup.com/jrsssa/article-abstract/135/3/370/7110572. Publisher: Oxford University Press.

Neyman, J., Dabrowska, D. M., and Speed, T. P. On the Application of Probability Theory to Agricultural Experiments. Essay on Principles. Section 9. *Statistical Science*, 5(4), November 1990. ISSN 0883-4237. doi: 10.1214/ss/1177012031. URL https://projecteuclid.org/journals/statistical-science/volume-5/issue-4/On-the-Application-of-Probability-Theory-to-Agricultural-Experiments-Essay/10.1214/ss/1177012031.full.

Nguyen, A.-p. and Martínez, M. R. On quantitative aspects of model interpretability, July 2020. URL http://arxiv.org/abs/2007.07584. arXiv:2007.07584 [cs, stat].

Oeing, J., Neuendorf, L. M., Bittorf, L., Krieger, W., and Kockmann, N. Flooding Prevention in Distillation and Extraction Columns with Aid of Machine Learning Approaches. *Chemie Ingenieur Technik*, 93(12):1917–1929, December 2021. ISSN 0009-286X, 1522-2640. doi: 10.1002/cite.202100051. URL https://onlinelibrary.wiley.com/doi/10.1002/cite.202100051.

Ohno, T. *Toyota production system: beyond large-scale production*. Productivity press, 2019. URL https://www.taylorfrancis.com/books/mono/10.4324/9780429273018/toyota-production-system-taiichi-ohno.

Oliveira, E. e., Miguéis, V., and Borges, J. Automatic root cause analysis in manufacturing: an overview & conceptualization. *Journal of Intelligent Manufacturing*, 34:1–18, February 2022. doi: 10.1007/s10845-022-01914-3.

Osborne, M. R., Presnell, B., and Turlach, B. A. A new approach to variable selection in least squares problems. *IMA journal of numerical analysis*, 20(3):389–403, 2000. URL https://academic.oup.com/imajna/article-abstract/20/3/389/777743. Publisher: Oxford University Press.

Pamfil, R., Sriwattanaworachai, N., Desai, S., Pilgerstorfer, P., Georgatzis, K., Beaumont, P., and Aragam, B. Dynotears: Structure learning from time-series data. In *International Conference on Artificial Intelligence and Statistics*, pp. 1595–1605. PMLR, 2020. URL http://proceedings.mlr.press/v108/pamfil20a.html.

Papadimitriou, S. and Mavroudi, S. The fuzzy frequent pattern tree. In *The WSEAS International Conference on Computers*, pp. 1–7, 2005. URL https://www.academia.edu/download/73752413/The_fuzzy_frequent_pattern_Tree20211028-17448-1hag0b3.pdf.

Papageorgiou, K., Theodosiou, T., Rapti, A., Papageorgiou, E. I., Dimitriou, N., Tzovaras, D., and Margetis, G. A systematic review on machine learning methods for root cause analysis towards zero-defect manufacturing. *Frontiers in Manufacturing Technology*, 2, 2022. ISSN 2813-0359. URL https://www.frontiersin.org/articles/10.3389/fmtec.2022.972712.

Park, H. and Jung, J.-Y. SAX-ARM: Deviant event pattern discovery from multivariate time series using symbolic aggregate approximation and association rule mining. *Expert Systems with Applications*, 141:112950, March 2020. ISSN 0957-4174. doi: 10.1016/j.eswa.2019.112950. URL https://www.sciencedirect.com/science/article/pii/S0957417419306682.

Pearl, J. Bayesian netwcrks: A model cf self-activated memory for evidential reasoning. In *Proceedings of the 7th conference of the Cognitive Science Society, University of California, Irvine, CA, USA*, pp. 15–17, 1985. URL http://ftp.cs.ucla.edu/pub/stat_ser/r43-1985.pdf.

Pearl, J. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan kaufmann, 1988. URL https://books.google.fr/books?hl=en&lr=&id=AvNID7LyMusC&oi=fnd&pg=PA1&dq=Pearl+J+(1988)+Probabilistic+reasoning+in+intelligent+systems:+networks+of+plausible+inference.+Morgan+Kaufmann,+Burlington&ots=F0VNVjlv54&sig=xjVbkmeWcMUtjKPwh9qoKMhDJl4.

Pearl, J. *Causality: Models, reasoning, and inference*. Causality: Models, reasoning, and inference. Cambridge University Press, New York, NY, US, 2000. ISBN 978-0-521-77362-1. Pages: xvi, 384.

Pearl, J. Causal inference in statistics: An overview. 2009a. URL https://projecteuclid.org/journals/statistics-surveys/volume-3/issue-none/Causal-inference-in-statistics-An-overview/10.1214/09-SS057.short.

Pearl, J. *Causality*. Cambridge University Press, September 2009b. ISBN 978-0-521-89560-6. Google-Books-ID: f4nuexsNVZIC.

Pearl, J. Structural Counterfactuals: A Brief Introduction. *Cognitive Science*, 37(6): 977–985, August 2013. ISSN 0364-0213, 1551-6709. doi: 10.1111/cogs.12065. URL https://onlinelibrary.wiley.com/doi/10.1111/cogs.12065.

Pearl, J. and Halpern, J. Y. Causes and Explanations: A Structural-Model Approach. Part I: Causes. *The British Journal for the Philosophy of Science*, 56(4):843–887, December 2005. ISSN 0007-0882, 1464-3537. doi: 10.1093/bjps/axi147. URL https://www.journals.uchicago.edu/doi/10.1093/bjps/axi147.

Pearl, J. and Mackenzie, D. *The book of why: the new science of cause and effect*. Basic books, 2018. URL https://scholar.google.com/scholar?cluster=25059012924853499 32&hl=en&oi=scholarr.

Pearson, K. LIII. On lines and planes of closest fit to systems of points in space. November 1901. doi: 10.1080/14786440109462720. URL https://zenodo.org/records/1430636.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Pei, J., Han, J., Lu, H., Nishio, S., Tang, S., and Yang, D. H-mine: 1st IEEE International Conference on Data Mining, ICDM'01. *Proceedings - 2001 IEEE International Conference on Data Mining, ICDM'01*, pp. 441–448, 2001. ISSN 0769511198. URL http://www.scopus.com/inward/record.url?scp=78149320187&partnerID=8YFLogxK.

Peiravan, H., Ilkhani, A. R., and Sarraf, M. J. Preventing of flooding phenomena on vacuum distillation trays column via controlling coking value factor. *SN Applied Sciences*, 2 (10):1670, September 2020. ISSN 2523-3971. doi: 10.1007/s42452-020-03470-y. URL https://doi.org/10.1007/s42452-020-03470-y.

Peters, J., Janzing, D., and Schölkopf, B. Causal Inference on Time Series using Restricted Structural Equation Models. In *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013. URL https://proceedings.neurips.cc/paper_files/paper/2013/hash/47d1e990583c9c67424d369f3414728e-Abstract.html.

Peters, J., Janzing, D., and Schölkopf, B. *Elements of causal inference: foundations and learning algorithms*. The MIT Press, 2017. URL https://library.oapen.org/handle/20.500.12657/26040.

Petersen, B. K. Deep symbolic regression. Technical report, Lawrence Livermore National Lab.(LLNL), Livermore, CA (United States), 2019. URL https://www.osti.gov/servlets/purl/1776654.

Petropoulos, F., Apiletti, D., Assimakopoulos, V., Babai, M. Z., Barrow, D. K., Ben Taieb, S., Bergmeir, C., Bessa, R. J., Bijak, J., Boylan, J. E., Browell, J., Carnevale, C., Castle, J. L., Cirillo, P., Clements, M. P., Cordeiro, C., Cyrino Oliveira, F. L., De Baets, S., Dokumentov, A., Ellison, J., Fiszeder, P., Franses, P. H., Frazier, D. T., Gilliland, M., Gönül, M. S., Goodwin, P., Grossi, L., Grushka-Cockayne, Y., Guidolin, M., Guidolin, M., Gunter, U., Guo, X., Guseo, R., Harvey, N., Hendry, D. F., Hollyman, R., Januschowski, T., Jeon, J., Jose, V. R. R., Kang, Y., Koehler, A. B., Kolassa, S., Kourentzes, N., Leva, S., Li, F., Litsiou, K., Makridakis, S., Martin, G. M., Martinez, A. B., Meeran, S., Modis, T., Nikolopoulos, K., Önkal, D., Paccagnini, A., Panagiotelis, A., Panapakidis, I., Pavía, J. M., Pedio, M., Pedregal, D. J., Pinson, P., Ramos, P., Rapach, D. E., Reade, J. J., Rostami-Tabar, B., Rubaszek, M., Sermpinis, G., Shang, H. L., Spiliotis, E., Syntetos, A. A., Talagala, P. D., Talagala, T. S., Tashman, L., Thomakos, D., Thorarinsdottir, T., Todini, E., Trapero Arenas, J. R., Wang, X., Winkler, R. L., Yusupova, A., and Ziel, F. Forecasting: theory and practice. *International Journal of Forecasting*, 38(3):705–871, July 2022. ISSN 01692070. doi: 10.1016/j.ijforecast.2021.11.001. URL https://linkinghub.elsevier.com/retrieve/pii/S0169207021001758.

Poursabzi-Sangdeh, F., Goldstein, D. G., Hofman, J. M., Wortman Vaughan, J. W., and Wallach, H. Manipulating and Measuring Model Interpretability. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pp. 1–52, Yokohama Japan, May 2021. ACM. ISBN 978-1-4503-8096-6. doi: 10.1145/3411764.3445315. URL https://dl.acm.org/doi/10.1145/3411764.3445315.

Preece, A., Harborne, D., Braines, D., Tomsett, R., and Chakraborty, S. Stakeholders in Explainable AI, September 2018. URL http://arxiv.org/abs/1810.00184. arXiv:1810.00184 [cs].

Quinlan, J. R. Induction of decision trees. *Machine Learning*, 1(1):81–106, March 1986. ISSN 0885-6125, 1573-0565. doi: 10.1007/BF00116251. URL http://link.springer.com/10.1007/BF00116251.

Quinlan, J. R. *C4.5: Programs for Machine Learning*. Elsevier, 1993. ISBN 978-0-08-050058-4. Google-Books-ID: b3ujBQAAQBAJ.

Radley-Gardner, O., Beale, H., and Zimmermann, R. (eds.). *Fundamental Texts On European Private Law*. Hart Publishing, 2016. ISBN 978-1-78225-864-3 978-1-78225-865-0 978-1-78225-866-7 978-1-78225-867-4. doi: 10.5040/9781782258674. URL http://www.bloomsburycollections.com/book/fundamental-texts-on-european-private-law-1.

Raissi, M., Perdikaris, P., and Karniadakis, G. E. Machine learning of linear differential equations using Gaussian processes. *Journal of Computational Physics*, 348:683–693, 2017. Publisher: Elsevier.

Raschka, S. MLxtend: Providing machine learning and data science utilities and extensions to Python's scientific computing stack. *Journal of Open Source Software*, 3(24):638, April 2018. ISSN 2475-9066. doi: 10.21105/joss.00638. URL http://joss.theoj.org/papers/10.21105/joss.00638.

Ribeiro, M. T., Singh, S., and Guestrin, C. " Why should i trust you?" Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1135–1144, 2016.

Rish, I. An empirical study of the naive Bayes classifier. In *IJCAI 2001 workshop on empirical methods in artificial intelligence*, volume 3, pp. 41–46, 2001. URL http://www.cc.gatech.edu/home/isbell/classes/reading/papers/Rish.pdf. Issue: 22.

Ross, S. and Bagnell, D. Efficient reductions for imitation learning. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pp. 661–668. JMLR Workshop and Conference Proceedings, 2010.

Ross, S., Gordon, G., and Bagnell, D. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 627–635. JMLR Workshop and Conference Proceedings, 2011.

{Royal Society}. Machine learning: the power and promise of computers that learn by example. *MACHINE LEARNING*, 2017.

Rubin, D. B. Estimating causal effects of treatments in randomized and nonrandomized studies. *Journal of Educational Psychology*, 66(5):688–701, October 1974. ISSN 1939-2176, 0022-0663. doi: 10.1037/h0037350. URL http://doi.apa.org/getdoi.cfm?doi=10.1037/h0037350.

Rudin, C. Stop Explaining Black Box Machine Learning Models for High Stakes Decisions and Use Interpretable Models Instead. *Nature machine intelligence*, 1(5):206–215, May 2019. ISSN 2522-5839. doi: 10.1038/s42256-019-0048-x. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9122117/.

Rudin, C., Chen, C., Chen, Z., Huang, H., Semenova, L., and Zhong, C. Interpretable machine learning: Fundamental principles and 10 grand challenges. *Statistics Surveys*, 16(none):1–85, January 2022. ISSN 1935-7516. doi: 10.1214/21-SS133. URL https://projecteuclid.org/journals/statistics-surveys/volume-16/issue-none/Interpretable-machine-learning-Fundamental-principles-and-10-grand-challenges/10.1214/21-SS133.full. Number: none Publisher: Amer. Statist. Assoc., the Bernoulli Soc., the Inst. Math. Statist., and the Statist. Soc. Canada.

Rudy, S. H., Brunton, S. L., Proctor, J. L., and Kutz, J. N. Data-driven discovery of partial differential equations. *Science Advances*, 3(4):e1602614, April 2017. doi: 10.1126/sciadv.1602614. URL https://www.science.org/doi/full/10.1126/sciadv.1602614. Publisher: American Association for the Advancement of Science.

Runge, J., Sejdinovic, D., and Flaxman, S. Detecting causal associations in large nonlinear time series datasets. *Science Advances*, 5, February 2017. doi: 10.1126/sciadv.aau4996.

Runkle, J. D., Sugg, M. M., Leeper, R. D., Rao, Y., Matthews, J. L., and Rennie, J. J. Short-term effects of specific humidity and temperature on COVID-19 morbidity in select US cities. *Science of the Total Environment*, 740:140093, 2020. URL https://www.sciencedirect.com/science/article/pii/S0048969720336135. Publisher: Elsevier.

Russell, S. J. and Norvig, P. *Artificial intelligence a modern approach*. London, 2010. URL https://ds.amu.edu.et/xmlui/bitstream/handle/123456789/10406/artificial%20intelligence%20-%20a%20modern%20approach%20%283rd%2C%202009%29.pdf?sequence=1&isAllowed=y.

Sabri, N. Fuzzy inference system: Short review and design. *Source of the DocumentInternational Review of Automatic Control*, January 2013.

Saepullah, A. and Wahono, R. S. Comparative analysis of mamdani, sugeno and tsukamoto method of fuzzy inference system for air conditioner energy saving. *Journal of Intelligent Systems*, 1(2):143–147, 2015. URL http://download.garuda.kemdikbud.go.id/article.php?article=360784&val=7117&title=Comparative%20Analysis%20of%20Mamdani%20Sugeno%20and%20Tsukamoto%20Method%20of%20Fuzzy%20Inference%20System%20for%20Air%20Conditioner%20Energy%20Saving. Publisher: IlmuKomputer. com.

Salinas, D., Flunkert, V., Gasthaus, J., and Januschowski, T. DeepAR: Probabilistic forecasting with autoregressive recurrent networks. *International Journal of Forecasting*, 36(3):1181–1191, 2020. Publisher: Elsevier.

Salmon, W. C. Causality without counterfactuals. *Philosophy of Science*, 61(2):297–312, 1994. URL https://www.cambridge.org/core/journals/philosophy-of-science/article/causality-without-counterfactuals/6BD80BF55BDA64FE2564D2472363BF8A. Publisher: Cambridge University Press.

Sayed, M. S. and Lohse, N. Distributed Bayesian diagnosis for modular assembly systems—A case study. *Journal of Manufacturing Systems*, 32(3):480–488, July 2013. ISSN 0278-6125. doi: 10.1016/j.jmsy.2013.03.001. URL https://www.sciencedirect.com/science/article/pii/S0278612513000290.

Schmidt, M. and Lipson, H. Distilling Free-Form Natural Laws from Experimental Data. *Science*, 324(5923):81–85, April 2009. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.1165893. URL https://www.science.org/doi/10.1126/science.1165893.

Schmidt, P. and Biessmann, F. Quantifying Interpretability and Trust in Machine Learning Systems, January 2019. URL http://arxiv.org/abs/1901.08558. arXiv:1901.08558 [cs, stat].

Schwartz, J. The distributed lag between air pollution and daily deaths. *Epidemiology*, 11(3): 320–326, 2000. URL https://journals.lww.com/epidem/Fulltext/2000/05000/The_Distributed_Lag_between_Air_Pollution_and.16.aspx. Publisher: LWW.

Schwarz, G. Estimating the dimension of a model. *The annals of statistics*, pp. 461–464, 1978. URL https://www.jstor.org/stable/2958889. Publisher: JSTOR.

Scutari, M., Graafland, C. E., and Gutiérrez, J. M. Who Learns Better Bayesian Network Structures: Constraint-Based, Score-based or Hybrid Algorithms? In *Proceedings of the Ninth International Conference on Probabilistic Graphical Models*, pp. 416–427. PMLR, August 2018. URL https://proceedings.mlr.press/v72/scutari18a.html. ISSN: 2640-3498.

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pp. 618–626, 2017. URL http://openaccess.thecvf.com/content_iccv_2017/html/Selvaraju_Grad-CAM_Visual_Explanations_ICCV_2017_paper.html.

Shalev-Shwartz, S. and Kakade, S. M. Mind the duality gap: Logarithmic regret algorithms for online optimization. *Advances in Neural Information Processing Systems*, 21, 2008.

Shimizu, S., Hoyer, P. O., Hyvärinen, A., Kerminen, A., and Jordan, M. A linear non-Gaussian acyclic model for causal discovery. *Journal of Machine Learning Research*, 7 (10), 2006. URL https://www.jmlr.org/papers/volume7/shimizu06a/shimizu06a.pdf?ref=https://codemonkey.link.

Shmueli, G. To explain or to predict? 2010. URL https://projecteuclid.org/journals/statistical-science/volume-25/issue-3/To-Explain-or-to-Predict/10.1214/10-STS330.short.

Shojaie, A. and Fox, E. B. Granger Causality: A Review and Recent Advances. *Annual Review of Statistics and Its Application*, 9(1):289–319, 2022. doi: 10.1146/annurev-statistics-040120-010930. URL https://doi.org/10.1146/annurev-statistics-040120-010930. Number: 1 _eprint: https://doi.org/10.1146/annurev-statistics-040120-010930.

Shortliffe, E. H. and Buchanan, B. G. A model of inexact reasoning in medicine. *Mathematical biosciences*, 23(3-4):351–379, 1975. URL https://www.sciencedirect.com/science/article/pii/0025556475900474. Publisher: Elsevier.

Silander, T., Leppä-Aho, J., Jääsaari, E., and Roos, T. Quotient normalized maximum likelihood criterion for learning Bayesian network structures. In *International conference on artificial intelligence and statistics*, pp. 948–957. PMLR, 2018. URL https://proceedings.mlr.press/v84/silander18a.html.

Sims, C. A. Macroeconomics and reality. *Econometrica: journal of the Econometric Society*, pp. 1–48, 1980. Publisher: JSTOR.

Slack, D., Friedler, S. A., Scheidegger, C., and Roy, C. D. Assessing the Local Interpretability of Machine Learning Models, August 2019. URL http://arxiv.org/abs/1902.03501. arXiv:1902.03501 [cs, stat].

Solé, M., Muntés-Mulero, V., Rana, A. I., and Estrada, G. Survey on Models and Techniques for Root-Cause Analysis, July 2017. URL http://arxiv.org/abs/1701.08546. arXiv:1701.08546 [cs].

Spirtes, P. and Glymour, C. An Algorithm for Fast Recovery of Sparse Causal Graphs. *Social Science Computer Review*, 9(1):62–72, April 1991. ISSN 0894-4393, 1552-8286. doi: 10.1177/089443939100900106. URL http://journals.sagepub.com/doi/10.1177/089443939100900106.

Spirtes, P., Glymour, C., and Scheines, R. *Causation, Prediction, and Search*, volume 81 of *Lecture Notes in Statistics*. Springer, New York, NY, 1993. ISBN 978-1-4612-7650-0 978-1-4612-2748-9. doi: 10.1007/978-1-4612-2748-9. URL http://link.springer.com/10.1007/978-1-4612-2748-9.

Spirtes, P., Glymour, C., and Scheines, R. *Causation, Prediction, and Search, 2nd Edition*, volume 1 of *MIT Press Books*. The MIT Press, September 2001. ISBN ARRAY(0x483e72e8). URL https://ideas.repec.org/b/mtp/titles/0262194406.html.

Srinivasan, R. and Chander, A. Explanation perspectives from the cognitive sciences—A survey. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pp. 4812–4818, 2021. URL https://www.researchgate.net/profile/Luis-Lamb/publication/342798575_Graph_Neural_Networks_Meet_Neural-Symbolic_Computing_A_Survey_and_Perspective/links/5f58d2ff299bf13a31adb1de/Graph-Neural-Networks-Meet-Neural-Symbolic-Computing-A-Survey-and-Perspective.pdf.

Stevens, J. A., Corso, P. S., Finkelstein, E. A., and Miller, T. R. The costs of fatal and non-fatal falls among older adults. *Injury prevention*, 12(5):290, 2006. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2563445/. Publisher: BMJ Publishing Group.

Strobl, E. and Lasko, T. A. Sample-Specific Root Causal Inference with Latent Variables. In *Proceedings of the Second Conference on Causal Learning and Reasoning*, pp. 895–915. PMLR, August 2023. URL https://proceedings.mlr.press/v213/strobl23b.html. ISSN: 2640-3498.

Sugeno, M. *Industrial applications of fuzzy control*. Elsevier Science Inc., 1985. URL https://dl.acm.org/doi/abs/10.5555/537323.

Sun, F., Liu, Y., Wang, J.-X., and Sun, H. Symbolic Physics Learner: Discovering governing equations via Monte Carlo tree search. February 2023. URL https://openreview.net/forum?id=ZTK3SefE8_Z.

Suzuki, J. Learning Bayesian belief networks based on the minimum description length principle: basic properties. *IEICE transactions on fundamentals of electronics, communications and computer sciences*, 82(10):2237–2245, 1999. URL https://search.ieice.org/bin/summary.php?id=e82-a_10_2237. Publisher: The Institute of Electronics, Information and Communication Engineers.

Taieb, S. B. and Hyndman, R. J. *Recursive and direct multi-step forecasting: the best of both worlds*, volume 19. Department of Econometrics and Business Statistics, Monash Univ., 2012. URL https://www.monash.edu/business/ebs/research/publications/ebs/wp19-12.pdf.

Tan, P.-N., Steinbach, M., and Kumar, V. *Introduction to data mining*. Always learning. Pearson, Harlow, new internat. edition edition, 2014. ISBN 978-1-292-02615-2.

Tan, S., Caruana, R., Hooker, G., and Lou, Y. Distill-and-Compare: Auditing Black-Box Models Using Transparent Model Distillation. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pp. 303–310, New Orleans LA USA, December 2018. ACM. ISBN 978-1-4503-6012-8. doi: 10.1145/3278721.3278725. URL https://dl.acm.org/doi/10.1145/3278721.3278725.

Telford, T. Apple Card algorithm sparks gender bias allegations against Goldman Sachs. *Washington Post*, 11, 2019. URL http://www.cs.williams.edu/~andrea/cs374/Articles/AppleCardWashingtonPost.pdf.

Tibshirani, R. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 58(1):267–288, 1996. URL https://academic.oup.com/jrsssb/article-abstract/58/1/267/7027929. Publisher: Oxford University Press.

Tibshirani, R., James, G., Witten, D., and Hastie, T. *An Introduction to Statistical Learning: with Applications in R*. Springer Texts in Statistics. Springer US, New York, NY, 2021. ISBN 978-1-07-161417-4 978-1-07-161418-1. doi: 10.1007/978-1-0716-1418-1. URL https://link.springer.com/10.1007/978-1-0716-1418-1.

Toti, G., Vilalta, R., Lindner, P., Lefer, B., Macias, C., and Price, D. Analysis of correlation between pediatric asthma exacerbation and exposure to pollutant mixtures with association rule mining. *Artificial intelligence in medicine*, 74:44–52, 2016.

Tsamardinos, I., Brown, L. E., and Aliferis, C. F. The max-min hill-climbing Bayesian network structure learning algorithm. *Machine Learning*, 65(1):31–78, October 2006. ISSN 0885-6125, 1573-0565. doi: 10.1007/s10994-006-6889-7. URL http://link.springer.com/10.1007/s10994-006-6889-7.

Tukey, J. W. *Exploratory data analysis*, volume 2. Reading, MA, 1977. URL http://theta.edu.pl/wp-content/uploads/2012/10/exploratorydataanalysis_tukey.pdf.

Udrescu, S.-M. and Tegmark, M. AI Feynman: A physics-inspired method for symbolic regression. *Science Advances*, 6(16):eaay2631, April 2020. ISSN 2375-2548. doi: 10.1126/sciadv.aay2631. URL https://www.science.org/doi/10.1126/sciadv.aay2631.

Uno, T., Kiyomi, M., and Arimura, H. LCM ver. 2: Efficient mining algorithms for frequent/closed/maximal itemsets. In *Fimi*, volume 126, 2004. URL http://www.philippe-fournier-viger.com/spmf/LCM2.pdf.

Van Beek, P. and Hoffmann, H.-F. Machine Learning of Bayesian Networks Using Constraint Programming. In Pesant, G. (ed.), *Principles and Practice of Constraint Programming*, volume 9255, pp. 429–445. Springer International Publishing, Cham, 2015. ISBN 978-3-319-23218-8 978-3-319-23219-5. doi: 10.1007/978-3-319-23219-5_31. URL https://link.springer.com/10.1007/978-3-319-23219-5_31. Series Title: Lecture Notes in Computer Science.

Van Lent, M., Fisher, W., and Mancuso, M. An explainable artificial intelligence system for small-unit tactical behavior. In *Proceedings of the national conference on artificial intelligence*, pp. 900–907. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2004. URL https://cdn.aaai.org/IAAI/2004/IAAI04-019.pdf.

Van Overschee, P. and De Moor, B. *Subspace identification for linear systems: Theory—Implementation—Applications*. Springer Science & Business Media, 2012.

Vegso, S., Cantley, L., Slade, M., Taiwo, O., Sircar, K., Rabinowitz, P., Fiellin, M., Russi, M., and Cullen, M. Extended work hours and risk of acute occupational injury: A case-crossover study of workers in manufacturing. *American Journal of Industrial Medicine*, 50(8):597–603, August 2007. ISSN 0271-3586, 1097-0274. doi: 10.1002/ajim.20486. URL https://onlinelibrary.wiley.com/doi/10.1002/ajim.20486.

Venkatraman, A., Boots, B., Hebert, M., and Bagnell, J. A. Data as demonstrator with applications to system identification. In *ALR Workshop, NIPS*, 2014.

Venkatraman, A., Hebert, M., and Bagnell, J. A. Improving multi-step prediction of learned time series models. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.

Vilone, G. and Longo, L. Notions of explainability and evaluation approaches for explainable artificial intelligence. *Information Fusion*, 76:89–106, 2021. URL https://www.sciencedirect.com/science/article/pii/S1566253521001093. Publisher: Elsevier.

Wang, J., Fujimaki, R., and Motohashi, Y. Trading Interpretability for Accuracy: Oblique Treed Sparse Additive Models. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '15, pp. 1245–1254, New York, NY, USA, August 2015. Association for Computing Machinery. ISBN 978-1-4503-3664-2. doi: 10.1145/2783258.2783407. URL https://doi.org/10.1145/2783258.2783407.

Wang, T. Gaining Free or Low-Cost Interpretability with Interpretable Partial Substitute. In *Proceedings of the 36th International Conference on Machine Learning*, pp. 6505–6514. PMLR, May 2019. URL https://proceedings.mlr.press/v97/wang19a.html. ISSN: 2640-3498.

Wang, X., Ding, H., Trajcevski, G., Scheuermann, P., and Keogh, E. Experimental Comparison of Representation Methods and Distance Measures for Time Series Data, December 2010. URL http://arxiv.org/abs/1012.2789. arXiv:1012.2789 [cs].

Wickham, H. Data Analysis. In Wickham, H. (ed.), *ggplot2: Elegant Graphics for Data Analysis*, Use R!, pp. 189–201. Springer International Publishing, Cham, 2016. ISBN 978-3-319-24277-4. doi: 10.1007/978-3-319-24277-4_9. URL https://doi.org/10.1007/978-3-319-24277-4_9.

Wiltz, C. Bias In, Bias Out: How AI Can Become Racist. 2017. URL https://www.designnews.com/artificial-intelligence/bias-in-bias-out-how-ai-can-become-racist.

Wipf, D. and Nagarajan, S. Iterative reweighted l1 and l2 methods for finding sparse solutions. *IEEE Journal of Selected Topics in Signal Processing*, 4(2):317–329, 2010. URL https://ieeexplore.ieee.org/abstract/document/5419071/. Publisher: IEEE.

Wold, S., Sjöström, M., and Eriksson, L. PLS-regression: a basic tool of chemometrics. *Chemometrics and intelligent laboratory systems*, 58(2):109–130, 2001. URL https://www.sciencedirect.com/science/article/pii/S0169743901001551. Publisher: Elsevier.

Wu, T. T. and Lange, K. Coordinate descent algorithms for lasso penalized regression. 2008. URL https://projecteuclid.org/journals/annals-of-applied-statistics/volume-2/issue-1/Coordinate-descent-algorithms-for-lasso/10.1214/07-AOAS147.short.

Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R., and Bengio, Y. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pp. 2048–2057. PMLR, 2015. URL https://proceedings.mlr.press/v37/xuc15.html.

Yao, L., Chu, Z., Li, S., Li, Y., Gao, J., and Zhang, A. A Survey on Causal Inference. *ACM Transactions on Knowledge Discovery from Data*, 15(5):1–46, October 2021. ISSN 1556-4681, 1556-472X. doi: 10.1145/3444944. URL https://dl.acm.org/doi/10.1145/3444944.

Yee, K., Tantipongpipat, U., and Mishra, S. Image Cropping on Twitter: Fairness Metrics, their Limitations, and the Importance of Representation, Design, and Agency. *Proceedings of the ACM on Human-Computer Interaction*, 5(CSCW2):1–24, October 2021. ISSN 2573-0142. doi: 10.1145/3479594. URL https://dl.acm.org/doi/10.1145/3479594.

Yuan, M. and Lin, Y. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 68(1):49–67, 2006. URL https://academic.oup.com/jrsssb/article-abstract/68/1/49/7110631. Publisher: Oxford University Press.

Yuan, Y. and Shaw, M. J. Induction of fuzzy decision trees. *Fuzzy Sets and systems*, 69(2): 125–139, 1995. URL https://www.sciencedirect.com/science/article/pii/016501149400229Z. Publisher: Elsevier.

Zadeh, L. A. Fuzzy sets. *Information and Control*, 8(3):338–353, June 1965. ISSN 0019-9958. doi: 10.1016/S0019-9958(65)90241-X. URL https://www.sciencedirect.com/science/article/pii/S001999586590241X.

Zaki, M. J. Scalable algorithms for association mining. *IEEE transactions on knowledge and data engineering*, 12(3):372–390, 2000. URL https://ieeexplore.ieee.org/abstract/document/846291/. Publisher: IEEE.

Zanga, A. and Stella, F. A Survey on Causal Discovery: Theory and Practice, May 2023. URL http://arxiv.org/abs/2305.10032. arXiv:2305.10032 [cs].

Završnik, A. Criminal justice, artificial intelligence systems, and human rights. *ERA Forum*, 20(4):567–583, March 2020. ISSN 1612-3093, 1863-9038. doi: 10.1007/s12027-020-00602-0. URL http://link.springer.com/10.1007/s12027-020-00602-0.

Zeiler, M. D. and Fergus, R. Visualizing and Understanding Convolutional Networks. In Fleet, D., Pajdla, T., Schiele, B., and Tuytelaars, T. (eds.), *Computer Vision – ECCV 2014*, volume 8689, pp. 818–833. Springer International Publishing, Cham, 2014. ISBN 978-3-319-10589-5 978-3-319-10590-1. doi: 10.1007/978-3-319-10590-1_53. URL http://link.springer.com/10.1007/978-3-319-10590-1_53. Series Title: Lecture Notes in Computer Science.

Zhang, C., Man, Z., and Nguyen, T. Analysis of 1: M conditional logistic regression modelling method. 1997. URL https://figshare.utas.edu.au/articles/conference_contribution/Analysis_of_1_M_conditional_logistic_regression_modelling_method/23079122/1. Publisher: University Of Tasmania.

Zhang, K. and Hyvarinen, A. On the Identifiability of the Post-Nonlinear Causal Model, May 2012. URL http://arxiv.org/abs/1205.2599. arXiv:1205.2599 [cs, stat].

Zhang, K. and Hyvärinen, A. Distinguishing Causes from Effects using Nonlinear Acyclic Causal Models. In *Proceedings of Workshop on Causality: Objectives and Assessment at NIPS 2008*, pp. 157–164. PMLR, February 2010. URL https://proceedings.mlr.press/v6/zhang10a.html. ISSN: 1938-7228.

Zhang, S., Li, X., Zong, M., Zhu, X., and Cheng, D. Learning k for kNN Classification. *ACM Transactions on Intelligent Systems and Technology*, 8(3):1–19, May 2017. ISSN 2157-6904, 2157-6912. doi: 10.1145/2990508. URL https://dl.acm.org/doi/10.1145/2990508.

Zhao, P. and Yu, B. On model selection consistency of Lasso. *The Journal of Machine Learning Research*, 7:2541–2563, 2006. URL https://www.jmlr.org/papers/volume7/zhao06a/zhao06a.pdf?ref=https://githubhelp.com. Publisher: JMLR. org.

Zheng, P., Askham, T., Brunton, S. L., Kutz, J. N., and Aravkin, A. Y. A unified framework for sparse relaxed regularized regression: SR3. *IEEE Access*, 7:1404–1423, 2018. URL https://ieeexplore.ieee.org/abstract/document/8573778/. Publisher: IEEE.

Zhong, J. and Negre, E. Ai: To interpret or to explain? In *Congrès Inforsid ((INFormatique des ORganisations et Systèmes d'Information et de Décision) 2021*, 2021. URL https://hal.science/hal-03529203/document.

Zhou, J., Gandomi, A. H., Chen, F., and Holzinger, A. Evaluating the quality of machine learning explanations: A survey on methods and metrics. *Electronics*, 10(5):593, 2021. URL https://www.mdpi.com/2079-9292/10/5/593. Publisher: MDPI.

Zio, E., Baraldi, P., and Popescu, I. C. A Fuzzy Decision Tree for Fault Classification. *Risk Analysis*, 28(1):49–67, February 2008. ISSN 0272-4332, 1539-6924. doi: 10.1111/j.1539-6924.2008.01002.x. URL https://onlinelibrary.wiley.com/doi/10.1111/j.1539-6924.2008.01002.x.

Zivot, E. and Wang, J. Vector Autoregressive Models for Multivariate Time Series. In *Modeling Financial Time Series with S-PLUS®*, pp. 385–429. Springer, New York, NY, 2006. ISBN 978-0-387-32348-0. doi: 10.1007/978-0-387-32348-0_11. URL https://doi.org/10.1007/978-0-387-32348-0_11.

ECOLE
DOCTORALE
DE MATHEMATIQUES
HADAMARD

Titre : Analyse Interprétable et Causale pour des Séries Temporelles Multivariées

Mots clés : Interprétabilité ; Causalité ; Séries temporelles; Apprentissage statistique

Résumé :
Les données de séries temporelles, qui mesurent l'évolution de variables au fil du temps, comme les relevés de capteurs, fournissent des informations précieuses sur le comportement des systèmes. En identifiant des structures dans ces données, nous pouvons comprendre les interactions entre les variables, améliorer la précision des prévisions et concevoir de meilleures stratégies d'intervention. Cette thèse étudie l'analyse de données de séries temporelles à haute dimension en se concentrant sur l'explication des déviations de systèmes par rapport à leur fonctionnement normal et sur la modélisation de la dynamique sous-jacente de systèmes permettant de prédire leur évolution. **Le premier objectif** est de développer un algorithme interprétable qui identifie les causes racines des comportements normaux et anormaux dans les données de séries temporelles. Diverses techniques sont utilisées pour identifier les causes racines, mais elles présentent des limites quant à leur capacité à traiter de grandes dimensions et à distinguer la causalité des corrélations. Une approche basée sur le concept de causalité de Granger [Granger 1988], qui extrait des relations interprétables et causales sous la forme de règles, a été développée pour remédier à ces limitations. L'algorithme qui en résulte est conçu pour traiter différents types de données (numériques, catégorielles), pour fournir aux utilisateurs des explications interprétables du problème et pour développer des règles prédictives permettant de désamorcer les phénomènes anormaux à l'avance.
Le deuxième objectif vise à développer un modèle de prévision qui non seulement prédit les valeurs futures, mais extrait également la dynamique sous-jacente des séries temporelles influençant ces prédictions. Ce domaine appelé régression symbolique favorise la transparence pour les utilisateurs en expliquant le raisonnement du modèle. Les modèles de régression avec pénalisation parcimonieuse sont largement utilisés dans ce domaine pour leur capacité à apprendre des dynamiques complexes dans des scénarios de grande dimension. Néanmoins, leurs performances en matière de prévision peuvent être limitées, en particulier pour des données complexes et non linéaires. Pour y remédier, nous proposons une nouvelle approche qui combine la régression pénalisée et la correction des erreurs dans un cadre de prévision des séries temporelles afin d'améliorer l'apprentissage des dynamiques sous-jacentes.

Title : Interpretable and Causal Analysis for Multivariate Time Series

Keywords : Interpretability; Causality; Time Series; Statistical learning

Abstract : Time-series data, which measure the evolution of variables over time, such as sensor readings, provide valuable information on the system's behavior. By identifying patterns in these data, we can understand the interactions between variables, improve forecasting accuracy, and design better intervention strategies. This thesis studies the analysis of high-dimensional time-series data, focusing on explaining local system deviations from normal operation and, on the global scale, modeling the underlying dynamics of the system to predict its evolution. **The first objective** is to develop an interpretable algorithm that identifies the root causes of both normal and abnormal behavior in time series data. Various techniques are used to identify root causes, but they suffer from limitations in their ability to handle high dimensions and to distinguish causality from correlations. To overcome these limitations, an approach based on the concept of Granger causality [Granger 1988], which extracts interpretable and causal relationships in the form of rules, has been developed. The resulting algorithm is designed to handle different data types (numerical, categorical), provide users with interpretable explanations of the problem, and develop predictive rules to defuse the event in advance.
The second objective aims at developping a forecasting model that not only predicts future values but also reveals the underlying dynamic of the time series influencing those predictions. This field, called symbolic regression, fosters transparency for users by explaining the model's reasoning. Regression models with sparse penalization are widely used in this field for their ability to learn complex dynamics in high-dimensional settings. Nevertheless, their forecasting performances can be limited, especially for complex and non-linear data. To address this, we propose a novel approach that combines penalized regression with forecasting error correction within a time series forecasting framework for improved learning of underlying dynamics.

Institut Polytechnique de Paris
91120 Palaiseau, France